[redacted]

Cisco® Field Manual: Catalyst® Switch Configuration

David Hucaby, CCIE® №4594
Stephen McQuerry, CCIE №6108

Cisco Press

201 West 103rd Street
Indianapolis, IN 46290 USA

Руководство Cisco® по конфигурированию коммутаторов Catalyst®

Руководство Cisco® по конфигурированию коммутаторов Catalyst®

Дэвид Хьюкаби, CCIE* №4594
Стив Мак-Квери, CCIE №6108

Москва • Санкт-Петербург • Киев
2004

ББК 32.973.26-018.2.75
Х98
УДК 681.3.07

Издательский дом "Вильямс"

Зав. редакцией *С.Н. Тригуб*

Перевод с английского *В.А. Шелепина*

Под редакцией *А.В. Мыльника*

По общим вопросам обращайтесь в Издательский дом "Вильямс" по адресу:
info@williamspublishing.com, http://www.williampublishing.com

Хьюкаби, Дэвид, **Мак-Клери**, Стив.

Х98 Руководство Cisco по конфигурированию коммутаторов Catalyst. : Пер. с англ. – М. : Издательский дом "Вильямс", 2004. – 560 с. : ил. – Парал. тит. англ.

ISBN 5-8459-0700-4 (рус.)

Эта книга — полное справочное руководство по всем функциям, поддерживаемым коммутаторами Catalyst. Она является продолжением опубликованной ранее книги по конфигурированию маршрутизаторов с характерным отличием, которое заключается в том, что различные операционные системы коммутаторов Catalyst для сравнения показаны вместе. В книге описан процесс конфигурирования коммутатора с использованием всех возможных вариантов команд разных операционных систем, используемых в устройствах Catalyst (CatOS и IOS).

ББК 32.973.26-018.2.75

Все названия программных продуктов являются зарегистрированными торговыми марками соответствующих фирм.

Никакая часть настоящего издания ни в каких целях не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами, будь то электронные или механические, включая фотокопирование и запись на магнитный носитель, если на то нет письменного разрешения издательства Cisco Press.

Authorized translation from the English language edition published by Cisco Press. Copyright © 2003

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Russian language edition published by Williams Publishing House according to the Agreement with R&I Enterprises International, Copyright © 2004

Книга выпускается при участии Официальной сетевой академии Cisco http://www.academy.cisco.com.

ISBN 5-8459-0700-4 (рус.)
ISBN 1-5870-5003-9 (англ.)

© Издательский дом "Вильямс", 2004
© Cisco Press, 2003

Оглавление

Содержание

Об авторах

Дэвид Хьюкаби (David Hucaby), сертифицированный эксперт CCIE № 4594, является ведущим сетевым инженером университета Кентукки, где занимается проблемами компьютерных сетей лечебных учреждений, основанных на коммутаторах Cisco Catalyst, IP-телефонией, линейками продуктов PIX и VPN-технологиями. До этого Дэвид занимал должность ведущего сетевого консультанта по вопросам проектирования и реализации сетей, уделяя основное внимание решениям для VPN сетей и IP-телефонии на основе оборудования корпорации Cisco. В университете Кентукки Дэвид получил степени бакалавра и магистра в области электротехники.

Стив Мак-Кюри (Steve McQuerry), сертифицированный эксперт CCIE № 6108 — инструктор и консультант с более чем десятилетним опытом работы в сетевой индустрии. Он является сертифицированным инструктором корпорации Cisco Systems (Certified Cisco Systems Instructor — CCSI), преподающим теоретические основы маршрутизации и коммутации для корпорации Global Knowledge. Стив также является разработчиком и руководителем курса *Advanced Cisco Campus Switching (Усовершенствованные методы коммутации в кампусах Cisco)* корпорации Global. Стив имеет степень бакалавра по инженерной физике, которую он получил в Восточном университете в Кентукки.

Технические рецензенты

Стефен А. Дилео (Stephen A. Dileo), президент компании Golden Networking Consultants Inc, является сетевым консультантом, среди клиентов которого — университет Южной Флориды в городе Санкт-Петербурге (St. Petersburg), медицинский округ Форт Лодердейла и Северного Бровард (Fort Lauderdale and North Broward Hospital District). Стефен является соавтором многих технических книг, изданных Cisco Press, а также действующим инструктором (CCSI № 97025), преподающим курсы BCMSN, BCRAN, CIPT, HSCN и ICND корпорации Cisco.

Мартин Дж. Дагган (Martin J. Duggan), сертифицированный эксперт CCIE № 7942, CCDP — технический разработчик корпорации IBM. Мартин занимает эту должность один год, а ранее он работал ведущим сетевым консультантом компании NTL. Специализация Мартина охватывает интеграцию крупномасштабных сетей, технологию ATM и проектирование локальных сетей территориального масштаба. Мартин проживает в Гемпшире, Великобритания, и получил квалификацию CCIE благодаря поддержке своей жены и детей — Анны и Джеймса.

Кевин Гамильтон (Kevin Hamilton), сертифицированный эксперт CCIE, работает в сетевой индустрии 20 лет, имеет дело с аналоговыми и цифровыми системами связи, включая широкополосные, Ethernet-, FDDI-, Token Ring- и ATM-сети. Он является соавтором книги *Принципы коммутации в локальных сетях Cisco (Cisco LAN Switching)*, изданной Cisco Press и ИД "Вильямс", а также составителем третьего издания *Руководства по технологиям объединенных сетей (Internetworking Technologies Handbook)*. Кевин ранее работал в компании Metsa Technologies на должности инструктора-консультанта, преподавая ATM, коммутацию, многоадресатный обмен данными, а также предметы, связанные с обеспечением безопасности сетей. В настоящее время Кевин обучает и консультирует персонал провайдеров служб, внедряющих технологии MPLS, многоадресатные технологии и сети на основе коммутации. Он имеет ученую степень Государственного университета штата Пенсильвания в области электротехники.

Джефф Тагг (Geoff Tagg) предоставляет консультации по проблемам сетей в Великобритании, имеет более чем двадцатилетний опыт работы с различными компаниями — от небольших предприятий до крупных многонациональных корпораций. До этого он был системным программистом и работал с множеством мэйнфреймов и миникомпьютеров. В настоящее время специализация Джеффа включает в себя технологии IP, ISDN, Frame Relay, ATM и Ethernet. Джефф вместе с семьей проживает в Оксфорде, Англия, и читает циклы лекций в университете Оксфорд Брукс (Oxford Brookes).

Посвящения

Дэвид Хьюкаби. Эта книга посвящается моей жене Марси и двум маленьким дочерям — Лорен и Карс, девочкам, никогда не включавшим коммутатор Catalyst, но неизменно вдохновлявшим меня с удовольствием продолжать работу. Я благодарен Богу, который дает мне терпение, бодрость духа (Римлянам 15:5) и возможность работать над этим и аналогичными проектами.

Стив Мак-Куерри. Я хочу посвятить эту книгу четырем наиболее важным в моей жизни людям. Бекки, ты — любовь всей моей жизни и мое вдохновение. Кэти, твое стремление сделать максимум возможного и твое отношение к делу всегда побуждают меня интенсивно работать. Логан, благодаря твоим вопросам и стремлению к знаниям я продолжаю постоянно учиться. Кэмерон, твоя энергия и сила духа делают меня молодым. Спасибо. Я не смог бы быть более довольным мужем, отцом или другом.

Благодарности

Дэвид Хьюкаби. Снова мне посчастливилось работать над книгой Cisco Press. Написание технической книги для меня очень увлекательно, хотя писать книгу исключительно во время обеденных перерывов и после того, как вся семья отправляется спать, тяжело. Я весьма признателен хорошим людям в издательстве Cisco Press за то, что они позволили мне работать над этим проектом, а также за их настойчивость, терпение и усердие при выполнении работы.

В частности, хотелось бы поблагодарить Брета Бартоу (Brett Bartow) за то, что он сделал данный проект осуществимым. Написание книги, подобной этой — длительный и трудный процесс. Брет всегда способен создать ощущение большой картины, одновременно заставляя следить за деталями. Я также весьма признателен за новое сотрудничество Крису Кливленду. Крис, вероятно, наиболее напряженно работающая личность из всех, кого я знаю, а также прекрасный редактор. Каким-то образом ему удается превратить "грубо обработанные" главы в гладкий текст.

Хотелось бы выразить признательность за напряженную работу и зоркие глаза нашим техническим рецензентам Стефену Далли, Мартину Даггану, Кевину Гамильтону и Джеффу Таггу. Настоящая книга представляет собой попытку объединить множество различных способов конфигурирования одинаковых функций на различных платформах Catalyst. Рецензенты выполнили поразительную работу, исправляя наши неточности и помогая нам организовать техническую информацию наилучшим образом. Я рад, что был не рецензентом, а автором. Хотелось бы выразить благодарность моему другу и спонсору Стиву Мак-Куерри. После Криса Стив — наиболее напри-

женно работающий из моих знакомых. Разделять с ним нагрузку по написанию книги для меня было большим удовольствием. Наконец, пора выразить признательность другу, который помог мне в моем личном "компьютерном кризисе". Я уронил свой ноутбук на бетонный пол. Жесткий диск, в отличие от остальных узлов ноутбука, удалось спасти. Я чрезвычайно благодарен моему другу Джошуа Фрайду (Joshua Fried), который нашел выход из этой ситуации, временно предоставив мне свой компьютер в тот же день, когда мой оказался неработоспособен. Благодаря его доброте мне удалось продолжить работу над книгой.

Стив Мак-Квэрри. Трудно поверить, что за три коротких года я получил возможность стать частью многих весьма увлекательных и смелых проектов. Поскольку много людей вместе создавали эту книгу, кажется несправедливым, что нам отводится только несколько строк для выражения им благодарности. С момента начала работы над книгой я твердо решил читать благодарности всех книг, которые попадают мне в руки, будь то техническая литература, художественная или иная. В издательской промышленности задействована большая группа людей, которые так же ответственны за публикуемые продукты, как и те, чьи имена значатся на обложках книг. Я бы хотел воспользоваться случаем, чтобы выразить индивидуальную благодарность всем, кто помогал мне в работе над моей частью этой книги.

Прежде всего хотелось бы поблагодарить моего друга и соавтора Дэвида Хьюкаби. Дэйв, я знаю, что ты делал гораздо больше своей части, и хочу поблагодарить тебя за потраченное тобой время, когда я брал на себя чрезмерные обязательства. За время моей карьеры мне трудно вспомнить кого-либо, работавшего с большим энтузиазмом и целенаправленностью, чем ты. В то же время более важным мне представляется твое внимание к Богу, семье и дружбе. Я счастлив быть твоим другом и партнером и надеюсь, что мы сможем продолжить наше сотрудничество в дальнейшем.

Как всегда, я хочу выразить благодарность Брету Бартоу. Я не думаю, что мы могли бы закончить книгу без его последовательности и обязательности. Спасибо за предоставленную возможность работать с тобой и за то, что ты поддерживал нашу мотивацию. Работать с тобой было истинным удовольствием.

Крис Кливленд, что я могу сказать! Ты был увлечен данным проектом и твой вклад — это большая часть того, что делает эту книгу цельным и полезным продуктом. Твои знания того, что мы пытаемся делать, и экспертная оценка как редактора неоценимы. Спасибо, что книга стала выглядеть лучше!

Нашим техническим редакторам — Кенину Гамильтону, Стефену Дэвису, Мартину Дагану и Джеффу Терри — спасибо за зоркие глаза и превосходные комментарии. Великолепно и то, что вы являлись частью команды.

Особая благодарность выражается прекрасным профессионалам Cisco Press: Джону Кейну (John Kane), Тамми Росс (Tammi Ross), Эми Льюис (Amy Lewis), Патрику Канноузу (Patrick Kanouse) и Эрику Шредеру (Eric Schroeder) — вы лучшие в индустрии!

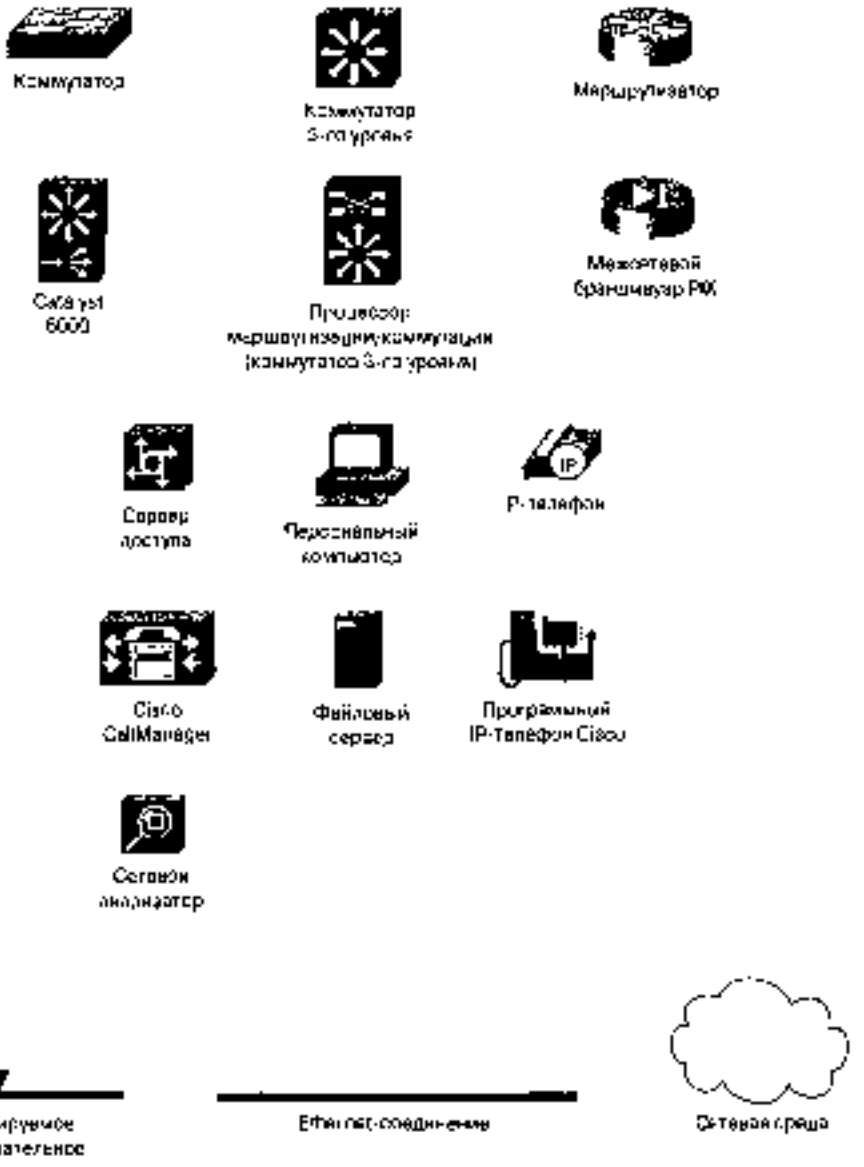
Спасибо всем моим студентам и коллегам-инструкторам в Global Knowledge. Ваша настойчивость и вопросы привели меня к лучшему пониманию материала.

Я хочу поблагодарить мою жену и детей за поддержку, которую они мне оказывают во всех моих проектах, а также за терпение и понимание в тех случаях, когда я допоздна задерживаюсь за работой и слегка раздражен на следующий день.

Самое важное, я благодарю Бога за то, что он дал мне знания, талант и возможность заниматься такой трудной и увлекательной работой.

Пиктограммы, используемые в этой книге

В книге используется большое количество пиктограмм для обозначения устройств корпорации Cisco, стандартных узлов сети, периферийного оборудования и других объектов. Ниже представлены пиктограммы с указанием объектов, которым они соответствуют.



Соглашения по синтаксису команд

Представленные ниже соглашения по синтаксису команд аналогичны соглашениям, используемым в *Справочнике по командам операционной системы IOS* (IOS Command Reference). В упомянутом справочнике используются следующие соглашения:

- с помощью вертикальной черты (|) разделяются альтернативные, взаимоисключающие элементы;
- в квадратных скобках [ ] указываются необязательные элементы;
- в фигурных скобках { } указываются необходимые элементы;
- **полужирным** шрифтом выделяются команды и ключевые слова, которые вводятся буквально, как показано; в примерах реальной конфигурации и сообщений системы (а также необычный синтаксис команд) с помощью жирного шрифта указываются команды, которые вводятся пользователем вручную (например, команда **show**);
- *курсивом* выделяются аргументы, для которых пользователь указывает реальные значения.

Введение

Это пособие является второй книгой в серии руководств корпорации Cisco, в которой основное внимание уделено линейке продуктов марки Catalyst. Из всего множества информационных источников и документации о коммутаторах Catalyst только несколько предоставляют краткие и компактные практические рекомендации для профессионалов сетевой индустрии.

Цель данной книги заключается в предоставлении краткого и простого справочного руководства по всем функциям, которые можно настроить на коммутаторах Catalyst. В сущности, тематика книги взята из целой серии программной документации коммутаторов Catalyst наряду с другими справочными материалами по сетевым технологиям и "сжата" в одну небольшую книгу, которую можно носить с собой. Кроме того, данная книга является единственной из имеющихся, в которой представлено конфигурирование функций коммутатора со всевозможными вариантами команд операционной системы, используемой в устройствах Catalyst (COS или CatOS и IOS).

Идея книги заключалась в продолжении опубликованной ранее книги по конфигурированию маршрутизаторов с характерным отличием, которое заключается в том, что различные операционные системы коммутаторов Catalyst для сравнения показаны вместе. В крупных коммутируемых сетевых средах часто наблюдается использование множества различных платформ Catalyst, в каждой из которых, возможно, имеется собственный набор функций и команды другой операционной системы. Как, вероятно, и многие специалисты, авторы находят трудным для запоминания этапы конфигурирования и команды, которые используются при переходе с одной платформы Catalyst на другую.

Как и в случае с конфигурацией маршрутизаторов, команды конфигурирования коммутаторов от руки записывались в блокнот, который сопутствовал автору при осуществлении практической деятельности сетевого консультанта и инженера. В ходе

работы часто требуется настроить какую-либо малознакомую функцию. В таких случаях хорошо иметь под рукой небольшой справочный блокнот. Эта книга вполне может послужить в роли такого компактного справочника.

Внимание!

Материал книги основан на наиболее современных на момент публикации версиях программного обеспечения Cisco Catalyst. COS-коммутаторы представлены согласно версии 7.2, а IOS-коммутаторы — согласно основной версии 12.2. Конфигурационные команды могут несколько отличаться при использовании более ранних версий программного обеспечения.

Особенности книги

Эта книга задумывалась как повседневный рабочий инструмент администратора сети, инженера или студента, и поэтому авторы избегали предоставления большого количества инструкций или теории по работе функций и команд, которые более подробно описаны в других книгах, охватывающих более узкую тематику.

Вместо этого книга заполнена главами, в которых кратко изложены факты, этапы конфигурации и пояснения к конфигурационным параметрам для каждой функции коммутатора Cisco Catalyst. Ниже кратко описана тематика глав книги.

Глава 1. "Использование интерфейса командной строки". В этой главе рассматриваются различия операционных систем COS и IOS, а также интерфейс командной строки.

Глава 2. "Функции коммутатора". В главе описаны LAN-коммутаторы и проектирование сети территориального масштаба.

Глава 3. "Конфигурирование блока Supervisor". В этой главе даны пояснения к конфигурированию приглашения командной строки коммутатора, IP-адреса, паролей, модулей коммутатора, управлению файлами, а также административных протоколов.

Глава 4. "Конфигурация интерфейсов второго уровня". В этой главе описывается конфигурирование интерфейсов Ethernet, Fast Ethernet, Gigabit Ethernet, EtherChannel, Token Ring и ATM LANE.

Глава 5. "Конфигурирование интерфейсов третьего уровня". В главе поясняется использование в коммутаторе интерфейсов третьего уровня.

Глава 6. "VLAN-сети и транкинг". В главе представлена VLAN-конфигурация, частные VLAN-сети, магистральные каналы, протокол VTP и динамический состав портов.

Глава 7. "Протокол распределенного связующего дерева (STP)". В главе обсуждаются операции, конфигурирование и тонкая настройка протокола STP.

Глава 8. "Многоуровневая коммутация". В главе освещается конфигурирование и использование аппаратного обеспечения коммутатора Catalyst для коммутации третьего уровня и обеспечения избыточности.

Глава 9. "Многоадресатные службы". В этой главе описана обработка многоадресатного трафика и взаимодействие с многоадресатными маршрутизаторами.

Глава 10. "Балансирование нагрузки на серверы (SLB)". В главе представлены функции коммутатора Catalyst 6000, которые упрощают доступ к группам серверов и межсетевым экранам.

Глава 11. "Управление трафиком и доступом к коммутатору". В главе рассматриваются способы подавления широковещания, фильтрации протоколов, аутентификации пользователей, защиты портов и списков VLAN доступа.

Глава 12. "Управление коммутаторами". В этой главе разъясняются такие вопросы, как настройка протоколов доступа к коммутаторам, SNMP- и RMON-администрирование, анализ портов SPAN, управление питанием и тестирование связи.

Глава 13. "Качество обслуживания". В главе представлены теоретические основы и функции конфигурирования QoS-параметров в коммутируемой сети.

Глава 14. "Поддержка передачи голосовых данных". В этой главе обсуждаются специализированные модули голосовых шлюзов, линейное питание, а также QoS-функции, необходимые для транспортировки голосового трафика.

Приложения A–F. В приложениях содержатся краткие справочные сведения о типах кабеля, стандартные порты и адреса, описываются специализированные модули и VLAN-расширения.

Как пользоваться книгой

Все сведения в этой книге приведены в соответствии с форматом краткого справочного руководства. Если известно, какую функцию или технологию необходимо использовать, можно перейти непосредственно к разделу, в котором она рассматривается. Нумерация разделов соответствует индексу краткого справочника, в котором представлены номера глав и разделов (например, номер 5.2 соответствует второму разделу пятой главы). Кроме того, на каждой странице расположена отличительная индексная метка, которая указывает на номер раздела.

Дополнительные сведения об особенностях книги

Каждый раздел главы начинается со сводного списка кратких сведений о рассматриваемой функции, технологии или протоколе. К таким спискам рекомендуется обращаться для изучения или обзора работы описываемой функции.

Этапы конфигурирования

Каждая функция, которая рассматривается в каком-либо разделе, включает в себя обязательные и необязательные команды, используемые для общей настройки. Различие заключается в том, что этапы конфигурирования представлены в общем виде. Следуя подобному описанию, можно настроить сложную функцию или технологию. Если выяснится, что нет необходимости использовать определенный параметр функции, данный урок и описание можно пропустить.

Кроме того, для каждого этапа конфигурации в описании совместно представлены команды как для операционной системы Catalyst (COS), так и для операционной системы IOS, поэтому можно изучать один и тот же раздел конфигурации независимо от используемого типа или модели коммутатора Catalyst.

Команды, отмеченные как "Система COS", относятся к операционной системе Catalyst, которая встречается в таких коммутаторах, как Catalyst 4000, 5000 и 6000 Supervisor 6000. Ниже следуют команды, отмеченные как "Система IOS", относящиеся к программному обеспечению операционной системы Cisco IOS, которая используется в коммутаторах Catalyst 2900XL, 2950, 3500XL, 3550, 4000 SupervisorIII и Catalyst 6000, работающих с "собственной гибкой операционной системой IOS", или "Supervisor IOS". В некоторых случаях "собственная", или "Supervisor IOS", отличается от IOS коммутаторов 2900/3500XL. Такие команды разделены в начальных этапах и сопровождаются соответствующими пояснениями там, где это необходимо.

Примеры конфигурирования функций

В каждом разделе приводится пример использования команд и их параметров. Команды в примерах представлены, как правило, в том порядке, в котором они вводились бы, следуя описанию. Часто бывает труднее изучать и понимать конфигурационные примеры из фактически используемых коммутаторов, поскольку команды отображаются в предопределенном порядке, а не в порядке их введения. Кроме того, примеры были очищены (там, где это возможно) с целью демонстрации только тех команд, которые приведены в разделе

Отображение сведений о функции

Там, где это практично, каждый раздел завершается кратким описанием команд, которые можно использовать для получения сведений об определенной функции коммутатора. Такие описания можно использовать в качестве краткого справочника при отладке или поиске неисправностей в работе коммутатора. Как и ранее, команды семейства show обеих операционных систем — как COS, так и IOS — представлены рядом для более простого использования.

Дополнительная литература

В конце каждой главы имеется список рекомендуемой литературы, где вы сможете найти более подробную информацию по рассматриваемой в главе теме.

От издательства

Вы, читатель этой книги, и есть главный ее критик и комментатор. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо, либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится или нет вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг. Наши координаты:

E-mail: `info@williamspublishing.com`
WWW: `http://www.williamspublishing.com`

Информация для писем из:

России: 115419, Москва, а/я 783
Украины: 03150, Киев, а/я 152

В этой главе...

- 1.1: **операционная система Catalyst (COS).** В этом разделе описывается применение операционной системы COS при конфигурировании коммутаторов.
- 1.2: **программное обеспечение межсетевой операционной системы Cisco (IOS).** В разделе описывается использование операционной системы Cisco IOS при конфигурировании коммутаторов.
- 1.3: **режим ROM-монитор.** В этом разделе описывается использование режима ROM-монитор для восстановления коммутатора и установки параметров загрузки.

Глава 1

Использование интерфейса командной строки

1.1: операционная система Catalyst (COS)

- В качестве интерфейса для конфигурирования и администрирования в системе COS (Catalyst Operating System — операционная система Catalyst) используется базовый интерпретатор командной строки.
- Операционная система COS выполняется на процессоре сетевого управления (Network Management Processor) коммутаторов Catalyst серий 4000, 5000 и 6000.
- К *интерфейсу командной строки (command-line interface — CLI)* можно получить доступ посредством консольного порта или с помощью Telnet-соединения.
- Для конфигурирования и администрирования в CLI-интерфейсе COS предусмотрено два режима — пользовательский (user mode) и привилегированный (privileged mode)
- Система COS не различает регистр символов и позволяет вводить сокращенные команды
- Справочная система и функции редактирования, имеющиеся в COS, упрощают использование этой операционной системы.
- Подсказки использования команд и их синтаксиса помогают пользователю устранить синтаксические ошибки
- В COS предусмотрены стандартные установки для большинства параметров коммутатора, которые позволяют упростить создание базовой конфигурации.

Использование операционной системы COS

В операционной системе COS имеются два основных режима администрирования коммутатора, а также множество полезных команд и параметров, которые предоставляют пользователю возможность в целях управления интерактивно взаимодействовать с коммутатором. Ниже описывается методика доступа к этим режимам и использование параметров для конфигурирования коммутатора.

1. Режимы пользовательского интерфейса командной строки.

   а) Режим непривилегированного пользователя.

   ```
   Cisco Systems Console
   Enter password: password
   Console>
   ```

   Пользователи могут подключаться к коммутатору посредством консольного порта или в сеансе Telnet. При первоначальном доступе к коммутатору пользователю предоставляется пользовательский режим (user mode) и ограниченное число команд. При подключении к коммутатору требуется *пароль (password)* пользовательского уровня. Стандартный пароль пользователя имеет *нулевую* длину, т.е. пароль не установлен, и для того чтобы войти в пользовательский режим, необходимо просто нажать клавишу <Enter>. В пользовательском режиме предоставляется доступ к командной строке, приглашение которой стандартно начинается со строки `Console>`. Для того чтобы закрыть соединение с консольным портом или Telnet-связь, следует ввести команду `logout` или `exit`.

   б) Привилегированный режим

   ```
   Console> enable
   Enter password: password
   Console> (enable)
   ```

   После того как пользователь получает доступ к пользовательскому режиму, появляется возможность использовать команду `enable` для входа в привилегированный (*privileged*), или так называемый "enable-режим", в котором предоставляется доступ ко всем командам и возможность установки параметров коммутатора. Для выхода из привилегированного EXEC-режима следует использовать команду `disable`. Кроме того, пользователь может закрыть соединение, воспользовавшись командой `logout` или `exit`.

2. Функции пользовательского интерфейса.

   а) Ввод команд

   ```
   Console> command
   Console> (enable) command
   ```

   Команды могут вводиться как из пользовательского, так и из привилегированного режимов. В пользовательском режиме доступна только команды получения сведений о коммутаторе и несколько тестовых утилит. Для того чтобы активизировать функцию или установить конфигурационный параметр коммутатора, необходимо, находясь в привилегированном режиме, использовать команды группы `set`. Для установки параметра следует написать команду и её опции, т.е. подставить значение вместо ключа *command*.

   Для того чтобы отключить сконфигурированные функции, следует использовать команду `set`, с помощью которой возвращаются стандартные параметры, или в некоторых случаях команду `clear`. Не для всех команд `set` имеется соответствующая команда `clear`. Для того чтобы просмотреть конфигурационные изменения, внесенные в интерфейсе системы COS, используется команда `show config` в привилегированном режиме. Чтобы просмотреть все установленные на коммутаторе параметры, нужно ввести команду `show`

config all. Кроме того, существует возможность просмотреть конфигурацию определенного модуля, используя команду **show config** *[mod]* **[all]**. Параметр **all** применяется для отображения как стандартных, так и нестандартных установок.

Редактировать командную строку можно с помощью клавиш управления курсором, которые используются для перемещения внутри командной строки. Редактирование осуществляется в режиме вставки. В данном случае это означает, что при вводе дополнительных символов оставшаяся с правой стороны часть командной строки сдвигается. Для внесения исправлений можно использовать клавиши <Backspace> и <Delete>.

б) Проверка синтаксиса.

Система COS представляет собой интерпретатор командной строки. Задача интерфейса заключается в получении от пользователя введенной командной строки, анализе данной строки и выполнении определенной функции по основании введенной команды. Если CLI-интерфейсу не удается распознать введенную пользователем команду, генерируется синтаксическая ошибка. *Синтаксическая ошибка (syntax error)* - сообщение, с помощью которого пользователь информируется о возникновении проблемы при вводе команды. Для COS синтаксическая ошибка, как правило, проявляется в одной из двух форм:

- Ошибка помощи.

  Ошибка помощи (help error) возникает в случае, когда интерфейс CLI не способен сопоставить пользовательский ввод с какой-либо корректной командой. Предположим, пользователь ввел команду **set speed port 3/5 100**. В таком случае генерируется ошибка помощи.

  ```
  Console> (enable) set speed port 3/5 100
  Unknown command "set speed". Use 'set help' for more info.
  ```

  (Неизвестная команда "set speed". Для получения дополнительной информации введите 'set help'.)

  Такое сообщение указывает пользователю на то, что система не может распознать ключевые слова, следующие за **set**. Если пользователь вводит команду **set help** или **set ?**, то будет отображен список всех возможных команд, начинающихся с ключевого слова **set**, среди которых имеется команда **set port speed**.

- Ошибка использования.

  Ошибка использования (usage error) возникает в том случае, если команда была введена верно, но имеется ошибка в выбранных для нее параметрах. Ошибка может возникнуть, если параметры не заданы, указаны в неверном порядке или не соответствуют данной команде. Ошибка использования генерируется в случае, если пользователь вводит команду **set port speed 100 3/5**.

  ```
  Console> (enable) set port speed 100 3/5
  Usage: set port speed <mod/port> <10|100|auto>
  ```

  Подобное сообщение указывает на то, что основная команда (**set port speed**) распознана, но ее параметры (**100** и **3/5**) некорректны. Отображая правило использования команды **set port speed** и все ее параметры, ин-

терфейс CLI информирует пользователя о правильном порядке параметров команды. Любые параметры, заключённые между угловыми скобками (< >), являются обязательными, тогда как все, указанные в квадратных скобках ([ ]), необязательны.

Совет

Иногда в процессе ввода пользователем какой-либо команды коммутатор отображает консольное сообщение. Часто пользователи нажимают клавишу <Enter>, для того чтобы избавиться от появившейся информации. Однако нажатие <Enter> в интерфейсе CLI приводит к тому, что интерпретатор анализирует команду или введенную часть команды (как правило, неполную), в результате чего возникает ошибка помощи или использования. Чтобы предотвратить возникновение подобной ситуации и обновить строку ввода, можно нажать комбинацию клавиш <Ctrl>+<R> или <Ctrl>+<L> для повторного отображения команды и продолжения редактирования.

н) Отображение конфигурации или информации.

```
Console> или Console> (enable) show [параметры]
```

Для отображения сведений о конфигурации или функционировании коммутатора используется команда show с соответствующим параметром. Стандартно отображаемая информация форматируется в блоки длиной 24 строки, т.е. после отображения 24-х строк вывод информации приостанавливается и появляется приглашение --More--. Чтобы продолжить отображение информации, нужно нажать клавишу <пробел> для отображения следующей страницы (24 строки) или <Enter> для построчного отображения оставшихся сведений. Для полного прекращения отображения информации используется клавиша <Q> или комбинация клавиш <Ctrl>+<C>.

п) Высота терминального экрана.

```
Console> (enable) set length lines
```

Приведенная команда применяется для изменения высоты блока выводимой информации до появления приглашения --More--. Если требуется предотвратить разделение информации на блоки, следует использовать команду set length 0. Можно отметить, что ширину экрана невозможно настроить в операционной системе COS, однако ее можно установить в терминальном приложении, которое используется для доступа к интерфейсу CLI. Отображаемых сведений может быть много, поэтому строки, длина которых превышает ширину экрана, как правило, разделяются терминальным приложением. В системе COS перенесенные строки не учитываются, поэтому верхняя часть страницы может перемещаться за верхнюю границу видимой области экрана в случае, если среди отображаемой информации имеются несколько разделенных строк.

Совет

Если с помощью команды show отображается большой блок информации, но необходимо найти определенный объект, то перемещаться по тексту можно путем введения косой черты (/), за которой вводится искомый текст (/текст). Если пользователь в приглашении --More-- вводит строку /текст, а затем нажимает клавишу <Enter>, то осуществляется поиск в отображаемой информации, и на экран возвращается часть текста, начинающаяся двумя строками выше строки, в которой содержится искомая строка. Если строка

текст не найдена, отображается сообщение "Pattern Not Found" (образец не найден). Кроме того, можно ввести букву **n** в приглашении --More-- для поиска последней введенной строки *текст*. Этот метод поиска может применяться во всех командах семейства **show**, в которых используется буфер для постраничного отображения информации, кроме команд **show cam**, **show mls** и **show tech-support**.

д) Контекстная справка.

Можно ввести знак вопроса (<?>) или ключевое слово **help** в любом месте командной строки для получения дополнительной информации от коммутатора. Если в командной строке введен только знак вопроса, то отображаются все доступные для данного режима команды. Знак вопроса может быть введен в любом месте после команды — например, вместо ключевого слова или параметра. Если за знаком вопроса вводится пробел, то отображаются все доступные ключевые слова или параметры. Если знак вопроса вводится без пробела за каким либо ключевым словом, то отображается список всех доступных команд, начинающихся с той же комбинации символов.

Кроме того, возможен ввод команд в сокращенном виде, которые сопровождается последующим нажатием клавиши <Tab>. При этом команда расширяется до своей полной формы, если не существует других команд, начинающихся с той же комбинации символов.

е) Журнал команд и повторный вызов команд.

В любом режиме ввода нажатием клавиши "стрелка вверх" (<↑>) или комбинации клавиш <Ctrl>+<P> вызывается предыдущая из ранее введенных команд. Аналогично при каждом нажатии стрелки вниз (<↓>) или комбинации клавиш <Ctrl>+<N> вызывается следующая из ранее введенных команд. После того как команда вызвана из журнала, ее можно редактировать так же, как и при обычном вводе. Команда **history** отображает дисциплинарный журнал команд (хронологический список). В коммутаторе в течение сеанса хранятся 20 последних команд. Для системы COS размер журнала не является конфигурируемым параметром.

Внимание!

Для клавиш управления курсором требуется ANSI-совместимый эмулятор терминала (VT100).

---

3. Сеансы Telnet-подключений

а) Начало нового сеанса

```
Console> (enable) telnet host
```

С помощью этой команды инициируется Telnet-подключение к узлу *host* (указанному с помощью IP-адреса или имени). После чего можно продолжить обмен данными с отдаленным узлом посредством CLI интерфейса коммутатора. На COS-устройствах одновременно Telnet-сеанс можно только из привилегированного режима.

б) Локальное завершение Telnet-сеанса.

В течение времени существования активного сеанса Telnet-подключения к узлу существует возможность вернуться к Telnet-приложению, нажав

комбинацию клавиш <Ctrl>+<]> (т.е. удерживая клавишу <Ctrl>, необходимо нажать правую квадратную скобку <]>). В таком случае Telnet-сеанс приостанавливается и пользователь возвращается к приглашению локального для коммутатора Telnet-приложения `telnet>`, где можно ввести команду `quit` для завершения Telnet-соединения, вместо того чтобы выходить с использованием удаленной системы. Такая возможность полезна в том случае, если Telnet-сеанс "блокируется" (locked up) и отсутствует ответ отдаленного узла.

b) Настройка таймаута сеанса.

```
Console (enable) set logout time
```

Все активные сеансы на коммутаторе, установленные как посредством консоли, так и через Telnet-соединения, завершаются (logged out) в случае, если в них отсутствует активность в течение определенного времени, которое определяется (в минутах) параметром *time* приведенной выше команды. Стандартное значение данного параметра равно двадцати минутам. Указанная ниже команда применяется для изменения значения таймаута закрытия. Если значение устанавливается равным 0, то сеансы не будут закрываться по таймауту.

1.2: программное обеспечение межсетевой операционной системы Cisco (IOS)

- Программное обеспечение Cisco IOS поддерживает доступ пользователей через CLI-интерфейс либо посредством web-браузера.
- Доступ к CLI-интерфейсу можно получить через консольный порт либо посредством сеанса Telnet.
- Запускать на выполнение команды Cisco IOS пользователи могут либо из *пользовательского уровня (user level)*, либо из *привилегированного уровня (privileged level)* интерфейса. На пользовательском уровне предоставляется основная системная информация и команды дистанционного подключения. Привилегированный уровень предоставляет полный доступ ко всей информации коммутатора, возможность изменения конфигурации, а также отладочные команды.
- Программное обеспечение Cisco IOS предоставляет пользователю множество уровней конфигурационных режимов, которые позволяют изменять конфигурацию различных ресурсов коммутатора.
- В операционной системе Cisco IOS предоставляется режим настройки базы данных VLAN-сетей для конфигурирования и модификации информации VLAN-сети и протокола VTP (VLAN Trunking Protocol — протокол магистрали канала сетей VLAN).
- Контекстная справочная система (context-sensitive help system) предоставляет правила ввода и перечень команд в любом пользовательском режиме.
- Имеется возможность сохранения журнала выполненных команд Cisco IOS. Кроме того, командные строки можно редактировать и использовать повторно.

- Отображаемые командами сведения можно просматривать и фильтровать, в результате чего сокращается время поиска полезной информации.
- Могут быть установлены предпочтительные значения параметров CLI-подключения к коммутатору.

Использование программного обеспечения Cisco IOS

В операционной системе Cisco IOS имеются два основных пользовательских режима для администрирования коммутатора и несколько других режимов, позволяющих контролировать конфигурацию устройства. В дополнение к различным режимам программное обеспечение Cisco ISO обеспечивает такие функции, как интерактивная справка и редактирование командной строки, которые позволяют взаимодействовать с коммутатором в административных целях. В последующих разделах описывается методика доступа к этим режимам и использование параметров для конфигурирования коммутатора.

1. Режимы интерфейса пользователя.

   а) **Пользовательский EXEC-режим**

   ```
   Switch>
   ```

   Пользователям предоставляется возможность подключаться к коммутатору посредством консольного порта или Telnet-сеанса. Стандартно при первоначальном доступе к коммутатору пользователь входит в *пользовательский EXEC-режим (user EXEC)*, в котором предоставляется ограниченный набор команд. При подключении к коммутатору может потребоваться *пароль* пользовательского уровня.

   б) Привилегированный EXEC-режим.

   ```
   Switch> enable
   password: [password]
   Switch#
   ```

   После того как пользователь получает доступ к пользовательскому EXEC-режиму, можно применить команду **enable** для входа в *привилегированный EXEC-режим (privileged EXEC)*, который также называется *административным режимом (enable)*. Такой режим предоставляет полный доступ ко всем командам. Для того чтобы покинуть привилегированный EXEC-режим, используется команда **disable** или **exit**.

   в) Конфигурационный режим.

   ```
   Switch# configure terminal
   ```

   Войти в конфигурационный режим можно из привилегированного EXEC-режима. В режиме конфигурации можно вводить любые команды для настройки функций коммутатора, которые доступны в программном образе операционной системы IOS. В режиме конфигурации пользователь управляет активной памятью коммутатора. Каждый раз при вводе корректной команды в любом конфигурационном режиме и нажатии клавиши <Enter> память немедленно изменяется. Конфигурационный режим организован иерархически. Режим глобальной конфигурации (global configuration mode) предоставляет команды, которые влияют на коммутатор в целом. В режиме конфигурирова-

ния интерфейса (interface configuration mode) предоставляются команды, позволяющие настраивать интерфейсы коммутатора в зависимости от настраиваемого ресурса. Пользователю предоставлена возможность входить в какой-либо другой из множества конфигурационных режимов и выйти из него. Для перехода со специфического уровня конфигурирования на более общий вводится команда `exit`. Для того чтобы покинуть режим глобальной конфигурации и вернуться в привилегированный EXEC-режим, необходимо ввести команду `exit` в режиме глобальной конфигурации. Для того чтобы покинуть любой конфигурационный режим и вернуться в привилегированный EXEC-режим, применяется команда `end` или комбинация клавиш <Ctrl>+<z>.

г) Режим конфигурирования базы данных VLAN-сетей.

```
Switch# vlan database
Switch(vlan)#
```

Перейти в указанный режим можно из привилегированного EXEC-режима. После ввода команды появится приглашение режима конфигурирования базы данных VLAN-сетей (vlan database mode). В данном режиме с помощью команд `vlan` и/или `vtp` конфигурируются и модифицируются VLAN- и VTP-параметры. После внесения изменений в базу данных VLAN они не вступят в действие до тех пор, пока не будет введена команда `apply` для активизации изменений в базе данных или команда `exit`, которая позволяет активизировать изменения и покинуть режим. При вводе команды `apply` или `exit` в режиме базы данных VLAN-сетей увеличивается номер ревизии VTP-конфигурации. Прежде чем создавать, изменять или редактировать конфигурацию VLAN-сетей, следует сконфигурировать VTP-домен или установить прозрачный VTP-режим. Команда `abort` отменяет какие-либо сделанные изменения в базе данных и позволяет покинуть рассматриваемый режим конфигурирования. Кроме того, существует возможность просмотреть текущее состояние базы данных и предполагаемые изменения, используя команды группы `show`. С помощью команды `help` или знака вопроса (<?>) в данном режиме доступна контекстная справка.

2. Функции интерфейса пользователя.

а) Ввод команд.

```
Switch>, Switch#, Switch(config)# или Switch(vlan)# command
Switch>, Switch#, Switch(config)# или Switch(vlan)# no command
```

Команды могут вводиться из любого режима (режим EXEC, глобальной конфигурации, конфигурации интерфейса, подынтерфейса, режим конфигурирования базы данных VLAN-сетей и т.д.). Для активизации какой-либо функции или параметра следует напечатать команду и обычно ее опции, т.е. параметр `command`. Для отключения действующей команды нужно ввести ключевое слово `no`, после которого следует название команды. Действующие настройки можно просмотреть с помощью команды `show running-config` в привилегированном режиме. Следует заметить, что некоторые команды и параметры устанавливаются стандартно и потому не отображаются в конфигурационном списке в виде командных строк.

Команды и параметры также можно сокращать и вводить в виде нескольких букв, количество которых достаточно для того, чтобы аббревиатура не была

двусмысленной. Например, для того чтобы войти в режим конфигурации интерфейса FastEthernet 0, можно ввести команду `interface fastethernet 0` в сокращенном виде: `int fa 0`.

Редактировать командную строку можно, перемещая курсор по строке с помощью клавиш "стрелка вправо" и "стрелка влево" (<^> и <.>). Если вводятся дополнительные символы, то оставшиеся справа знаки сдвигаются. Для внесения изменений можно использовать клавиши <Backspace> и <Delete>.

Внимание!

Если коммутатор при вводе команды отображает консольную информацию или сообщение об ошибке, для повторного вывода строки и продолжения редактирования можно использовать клавиши <Ctrl>+<L> или <Ctrl>+<R>. Кроме того, линии (консольные, VTY или aux) можно настроить на использование режима `logging synchronous`. В этом режиме коммутатор автоматически обновляет строки после вывода информации. Возможно, придется подождать пока коммутатор отобразит информацию: если вводятся команды `debug` с включенным режимом `logging synchronous`, может потребоваться некоторое время, пока коммутатор завершит выполнение команды (например, ping) и отобразит информацию.

б) Контекстная справка.

Чтобы получить от коммутатора дополнительную информацию, можно в любой позиции командной строки ввести знак вопроса (<?>). Если в строке напечатан только знак вопроса, будут отображены все возможные для данного режима команды. Знак вопроса может вводиться в любой позиции после команды, ключевого слова или параметра. Знак вопроса, введенный после пробела, позволяет отобразить все доступные ключевые слова или параметры. Если знак вопроса вводится без пробела после другого слова, отображаются все доступные команды, начинающиеся с данной подстроки. Эта функция может быть полезной в случае, когда сокращенная команда является двусмысленной и отмечается как ошибочная.

Сокращенные команды также можно вводить с последующим нажатием клавиши <Tab>. Название команды расширяется до ее полной формы, если сокращение не является двусмысленным.

В ситуации, когда командная строка вводится с некорректным синтаксисом, возвращается сообщение об ошибке "`% Invalid input detected at '^' marker`" (обнаружена ошибка в позиции маркера '^'). Знак вставки (^) появляется ниже соответствующего символа командной строки в той позиции, где обнаружена синтаксическая ошибка.

в) Хронологический список команд.

- Настройка количества сохраняемых команд (стандартно 10) — это *необязательная функция*.

  Для того чтобы установить размер журнала команд для текущего терминального сеанса, необходимо ввести следующую команду:

  `Switch# terminal history [size <lines>]`

  Для установки размера журнала всех сеансов по линии используется следующая команда:

```
Switch(config-line)# history [size lines]
```

- Повторный вызов введенных команд.

  В любом режиме работы интерфейса командной строки каждое нажатие стрелки вверх (<^>) или комбинации клавиш <Ctrl>+<P> повторно вызывает предыдущую введенную команду. Аналогично при каждом нажатии стрелки вниз (<v>) или комбинации клавиш <Ctrl>+<N> повторно вызывается следующая из ранее введенных команд. Вызванные из хронологического списка команды можно редактировать как введённые с клавиатуры. С помощью команды **show history** отображается журнал записанных команд.

Внимание!

Для клавиш управления курсором требуется ANSI-совместимый эмулятор терминала (VT100).

г) Поиск и фильтрация сведений, отображаемых командами

- Поиск и фильтрация сведений, предоставленных командой **show**.

  ```
  Switch# show command ... | {begin  include | exclude} reg-
  expression
  ```

  Команда **show** способна генерировать длинные отчеты. Если листинг содержит большее количество строк, чем может быть отображено в терминальном сеансе (что устанавливается с помощью параметра *length*), то информация отображается до заполнения экрана. При этом внизу появляется приглашение **--More--**. Для просмотра следующего экрана вывода необходимо нажать клавишу пробела. Для отображения одной дополнительной строки используется клавиша <Enter>. Для возврата в командную строку следует воспользоваться клавишей <Q> или комбинацией <Ctrl>+<C>, или любой другой клавишей, кроме пробела или <Enter>.

  Для того чтобы найти определенное регулярное выражение и начать вывод с данной строки, используется ключевое слово **begin**. Это может оказаться полезным в ситуации, когда в конфигурации коммутатора имеется множество интерфейсов. Например, чтобы построчно просматривать конфигурацию и в конечном итоге найти определенную строку, можно нажимать клавишу пробела, однако гораздо эффективнее будет использовать ключевое слово **begin** для немедленного перехода к желаемой строке. Для отображения только тех строк, которые включают в себя регулярное выражение, используется ключевое слово **include**. Чтобы вывести на экран все строки, не включающие в себя регулярное выражение, используется ключевое слово **exclude**.

- Анализ информации, предоставленной командой **more**.

  ```
  Switch# more file-url | {begin | include | exclude}
  reg-expression
  ```

  С помощью команды **more** отображается содержимое какого-либо файла на коммутаторе. Как правило, данная команда используется для отображения файла первоначальной загрузки (**more nvram:startup-config**) или файла действующей конфигурации (**more system:running-config**). Стандартно файл отображается постранично с приглашением **--More--**.

Для того чтобы найти определенное регулярное выражение и начать вывод с данной строки, используется ключевое слово `begin`. Для отображения только тех строк, которые включают в себя регулярное выражение, используется ключевое слово `include`. Чтобы вывести на экран все строки, не включающие в себя регулярное выражение, используется ключевое слово `exclude`.

- Поиск в приглашении --More--.

```
| more  ||/ | + | -|regular-expression
```

В приглашении --More-- поиск можно осуществлять путем ввода символа косой черты (</>), за которой следует регулярное выражение. Для отображения только тех строк, в которых содержится данное регулярное выражение, следует нажать клавишу "плюс" (<+>). Чтобы вывести только строки, в которых регулярное выражение отсутствует, используется клавиша "минус" (<->).

- Что такое регулярное выражение?

Регулярное выражение может применяться для сравнения со строками отображаемой информации. Регулярное выражение составляется из шаблонов: либо простых текстовых строк (*ethernet* или *oup*), либо более сложных шаблонов для сравнения. Как правило, регулярные выражения представляют собой обычные текстовые слова, позволяющие обнаружить место расположения искомой строки в информации, выводимой с помощью команды `show`.

Более сложные регулярные отражения составляются из шаблонов и операторов. В табл. 1.1 приведены символы, которые применяются в качестве операторов.

Таблица 1.1. Символы операторов

Символ	Значение
.	Соответствует одному символу
*	Соответствует нулю или нескольким предшествующим последовательным шаблонам
+	Соответствует одной или нескольким последовательностям предшествующего шаблона
?	Соответствует нулю или одному вхождению предшествующего шаблона
^	Совпадение в начале строки
$	Совпадение в конце строки
\	Соответствует запятой, скобкам, интервалам, началу или концу строки, пробелу
[]	Определяет в качестве шаблона диапазон символов
()	Группирует символы в качестве шаблона. Если скобки окружают какой-либо шаблон, то его позднее можно вызвать повторно в выражении путем использования обратной косой черты (<\>) с указанием количества вхождений

3. Терминальные сеансы.

а) Начало нового сеанса.

```
Switch# telnet host
```

С помощью этой команды инициируется Telnet-подключение к узлу *host* (указывается либо IP-адрес, либо имя узла). После чего обмен данными с отдаленным узлом может быть продолжен из CLI-интерфейса коммутатора.

б) Наименование сеанса.

```
Switch# name-connection
Switch# Connection number: number
Switch# Enter logical name: name
```

Активному сеансу может быть присвоено имя в виде текстовой строки, что упрощает идентификацию сеанса с помощью команд **show sessions** и **where**.

в) Приостановка сеансов для выполнения других задач.

Чтобы приостановить активный Telnet-сеанс, необходимо ввести управляющую **последовательность** (escape sequence) <Ctrl>+<Shift>+<6> с последующим нажатием клавиши <x> (т.е. необходимо нажать вместе клавиши <Ctrl>, <Shift> и <6>, отпустить их, а затем нажать букву <x> на клавиатуре). Последовательность перехода в режим ожидания иногда записывается как <Ctrl>—<^><x>. Она приостанавливает Telnet-сеанс и возвращает пользователя к приглашению командной строки локального коммутатора.

Внимание!

Возможна ситуация, при которой последовательно открыто несколько Telnet-сеансов. Например, с локального коммутатора можно подключиться к коммутатору А, а затем с него — к коммутатору Б и так далее. Для того чтобы приостановить один из таких сеансов, необходимо также соблюдать последовательности выхода. Ввод одной последовательности <Ctrl><^><x> приостанавливает сеанс подключения к коммутатору А и возвращает пользователя в командную строку локального коммутатора. Ввод последовательности <Ctrl><^><Ctrl><^><x> приостанавливает сеанс подключения к коммутатору Б и возвращает пользователя к приглашению коммутатора А. (Необходимо только ввести букву <x> в завершающей последовательности.)

г) Отображение всех активных сеансов

```
Switch# show sessions
```

Все открытые сеансы для данного подключения к локальному коммутатору перечисляются вместе с номерами соединений. Такую же информацию можно получить, воспользовавшись командой **where**.

д) Возврат к определенному сеансу.

Прежде всего необходимо с помощью команды **show sessions** определить номер соединения для требуемого сеанса. Затем в командной строке просто следует ввести полученный номер соединения, после чего сеанс будет восстановлен. Кроме того, можно просто нажать клавишу <Enter> или <Return> в приглашении командной строки, и последнее в списке активное соединение повторно активизируется. Последнее активное соединение в списке отмечается звездочкой (*). Указанные способы упрощают переключение между локальным коммутатором и одним отдаленным сеансом.

е) Окончание активного сеанса.

```
Switch2#<Ctrl><^> <x>
Switch1# disconnect connection-number
```

В ситуации, когда сеанс удаленного подключения приостановлен, для его окончания и закрытия Telnet-соединения используется команда disconnect. В противном случае сеанс останется открытым до тех пор, пока на отдаленном узле не истечет время ожидания (если оно предусмотрено).

Внимание!

При возобновлении соединений появляется сообщение "[Resuming connection 2 to switch ...]." (возобновление подключения 2 к коммутатору ...). После повторной активизации соединения сообщение, приведенное выше, не изменяется, а приглашение коммутатора не отображается. Поэтому, чтобы действительно возобновить соединение, необходимо снова нажать клавишу <Enter> для окончательного восстановления соединения и получения приглашения от требуемого устройства.

ж) Формат терминального экрана.

- Установка размера экрана только для текущего сеанса осуществляется с помощью следующих команд:

```
Switch# terminal length lines
Switch# terminal width characters
```

- Установка размера экрана для всех сеансов

```
Switch(config-line)# length lines
Switch(config-line)# width characters
```

Ширина экрана устанавливается равной количеству символов, указанному в параметре (characters), высота равна количеству строк (lines). Когда количество строк в выводе команды превышает параметр lines, используется приглашение --More--. Если постраничное отображение нежелательно, используется параметр length 0. Стандартная высота экрана для сеанса равна 24 строкам, а стандартная ширина — 80 символам.

з) Конфигурирование времени ожидания сеанса.

- Параметр абсолютного времени ожидания для линии устанавливается с помощью указанной ниже команды.

```
Switch(config-line)# absolute-timeout minutes
```

Все активные сеансы уничтожаются по истечении времени (в минутах), равного параметру minutes. (Стандартное значение равно 0 минут или не определенному времени ожидания завершения сеанса.)

- Параметр времени ожидания для неактивной линии указывается с помощью команды

```
Switch(config-line)# session-timeout minutes [output]
```

Все активные сеансы в линии прекращаются, только если они простаивают время, равное параметру minutes. (Стандартное значение равно 0 минут или неопределенному времени ожидания окончания сеанса.) Использование ключевого слова output приводит к тому, что таймер простоя перезапускается при возникновении исходящего трафика в линии, поддерживая активное соединение.

- Тайм-аут для всех сеансов EXEC-режима задается с помощью команды

```
Switch(config-line)# exec-timeout minutes [seconds]
```

Активные сеансы EXEC-режима автоматически закрываются после истечения таймаута простоя, выраженного в минутах (*minutes*) и секундах (*seconds*) (стандартное значение равно 10 минутам). Для отключения в данной линии таймеров времени простоя используется команда `no exec-timeout` или `exec-timeout 0 0`.

- Включение предупреждения об истечении времени ожидания для сеанса задается с помощью следующей команды:

```
Switch(config-line)# logout-warning [seconds]
```

Пользователи получают предупреждение об оставшемся времени в секундах (*seconds*) до приближающегося отключения. Стандартно предупреждение не передается. Если в поле *seconds* значение не указано, то принимается стандартное значение, равное 20 секундам.

4. Web-интерфейс коммутатора.

а) Включение Web-интерфейса.

```
Switch(config)# ip http server
```

С помощью этой команды запускается сервер Web-интерфейса, позволяющий пользователям осуществлять мониторинг или конфигурирование коммутатора посредством Web-браузера.

Внимание!

Ввиду значительной уязвимости служб HTTP-сервера использовать Web-интерфейс коммутатора для доступа из открытых сетей (таких, как Internet) не следует. Данная уязвимость документируется как Cisco Bug ID CSCdt93862 (дефект № CSCdt93862). Для отключения HTTP-сервера используется команда `no ip http server`. В дополнение к этой проблеме в стандартной аутентификации используются открытые текстовые пароли. Если возникла необходимость использовать Web-интерфейс, следует обеспечить конфигурирование более строгого метода аутентификации и ограничить доступ на этапах в HTTP, описания которых приведено ниже.

б) Установка номера порта Web-браузера (*необязательный этап*).

```
Switch(config)# ip http port number
```

HTTP-трафик может использовать TCP-порт, указанный в параметре *number* (стандартно используется порт с номером 80).

в) Ограничение доступа к Web-интерфейсу (*необязательный этап*).

```
Switch(config)# ip http access-class access-list
```

Стандартный список IP-доступа (определенный по номеру или по имени) может использоваться для ограничения значений IP-адресов отправителей для узлов, имеющих доступ к Web-интерфейсу. Подобное ограничение следует применить для того, чтобы сузить круг потенциальных пользователей, имеющих доступ к Web-интерфейсу коммутатора.

г) Выбор метода аутентификации пользователей (*необязательный этап*)

```
Switch(config)# ip http authentication {aaa | enable | local
tacacs}
```

У пользователей, пытающихся подключиться к Web-интерфейсу коммутатора, могут быть затребованы пароли и проведена аутентификация с помощью нескольких механизмов. Стандартом для аутентификации используется метод **enable** (необходимо ввести **enable**-пароль). Нужно использовать один из более строгих методов аутентификации: **aaa**, **local** (аутентификация осуществляется с использованием имен пользователей и паролей локально на коммутаторе) и **tacacs** (стандартная или расширенная TACACS-аутентификация).

а) Просмотр страницы конфигурирования коммутатора

Доступ к странице осуществляется из Web-браузера с использованием URL-адреса `http://switch/`, где параметру `switch` соответствует IP-адрес или имя коммутатора. Стандартная страница коммутатора доступна для пользователей с уровнем привилегии 15. Для пользователей, ограниченных уровнем привилегий ниже 15, доступны только IOS-команды низших уровней привилегий

1.3: режим ROM-монитор

- Режим ROM-монитор (ROM monitor mode) представляет собой ROM-программу, которая выполняется во время включения или перезапуска коммутатора.
- Получить доступ к интерфейсу ROM-монитора можно с помощью нажатия комбинации клавиш <Ctrl>+<Break> в процессе загрузки коммутатора.
- Если при загрузке операционной системы произошел сбои или в поле BOOT конфигурационного регистра (configuration register) установлено значение 0, то коммутатор входит в режим ROM-монитор.
- Если в коммутаторе возникает фатальная исключительная ситуация, из которой восстановление невозможно, то коммутатор также входит в режим ROM-монитор
- Как и в случае с операционными системами COS и IOS, ROM-монитор является CLI-интерфейсом.
- ROM-монитор предоставляет ограниченный набор команд, связанных с восстановлением коммутатора при загрузке.
- ROM-монитор предоставляет в помощь пользователю ограниченные справочные возможности и базовые функции журнала команд.
- ROM-монитор позволяет осуществлять асинхронную транспортировку данных по протоколу Xmodem в целях оказания пользователю помощи при восстановлении операционной системы

Использование набора команд режима ROM-монитор

Многие коммутаторы обладают набором команд ROM-монитор, который позволяет пользователю взаимодействовать с коммутатором при восстановлении операционных систем или изменении загрузочных переменных в процессе загрузки. Режим ROM-монитор в помощь пользователю предоставляет базовый набор команд и справочную службу. В последующих разделах описано использование возможностей режима ROM-монитор.

1. Режимы пользовательского интерфейса.

   rommon 1 >

   Интерфейс *rommon* представляет собой простой CLI интерфейс, позволяющий пользователю восстанавливать его после сбоев или изменять загрузочные параметры коммутатора. В этом интерфейсе имеется один режим с ограниченным набором команд, как правило, связанных с загрузкой коммутатора и управлением параметрами среды.

2. Функции пользовательского интерфейса.

   а) Ввод команд

      rommon> command

      Командная строка режима rommon>, как и CLI-интерфейс систем COS и IOS, интерпретирует за один раз одну введенную строку.

   б) Интерактивная справка.

      Для того чтобы получить список доступных команд режима rommon, можно ввести знак вопроса (?) в начале строки приглашения rommon>

   в) Журнал команд.

      Рассматриваемый интерфейс поддерживает хронологический список до семнадцати введенных ранее команд. Для просмотра списка используется команда `history` или вводится буква `h`. В выведенном списке для каждой команды можно увидеть числовое значение. Повторный вызов команды осуществляется с помощью команды `repeat value` или с `value`, где параметр `value` равен значению, указанному с левой стороны от каждой команды в хронологическом списке.

3. Проверка и изменение конфигурационных переменных.

   а) Проверка конфигурационных переменных.

      rommon> `set`

      В режиме ROM-монитор конфигурационные переменные коммутатора загружаются до того, как пользователи получат доступ к командной строке. Переменные включают в себя местоположение конфигурационного файла и загрузочного образа, которые будут использованы ROM-монитором. Для того чтобы просмотреть переменные, используется команда `set`.

   б) Установка конфигурационных переменных.

      rommon> `PARAMETER=value`

      Чтобы установить какую-либо конфигурационную переменную, используется название параметра, которое должно быть введено *точно так, как оно указано* в команде `set` (т.е. с соблюдением регистра символов), за которым следует значение. Для того чтобы обнулить конфигурационную переменную, следует оставить поле значения пустым. Например, для удаления ссылки на загрузочный образ, указанный для коммутатора, используется следующая команда.

      rommon> `BOOT=`

Внимание!

Важно отметить, что в режиме ROM-монитор все символы какой-либо переменной или параметра должны вводиться в верхнем регистре, а символы команд — в нижнем. Если символы будут введены в неверном регистре, ROM-монитор не сможет обработать команду.

в) Сохранение конфигурационных переменных.

```
rommon> sync
```

Чтобы сохранить конфигурационные переменные, используется команда sync, записывающая новые переменные в энергонезависимую память (NVRAM) для того, чтобы можно было их использовать после перезапуска устройства.

г) Загрузка новых конфигурационных переменных.

```
rommon> reset
```

Для того чтобы загрузить конфигурационные переменные в ROM-монитор, необходимо совершить цикл отключения питания или перезапустить коммутатор. Перезапуск устройства инициируется указанной выше командой `reset`.

4. Загрузка коммутатора в режиме ROM-монитор.

а) Просмотр образов на flash-устройствах хранения данных.

```
rommon> dir [device:]
```

На ROM-монитор возлагается ответственность за загрузку образа операционной системы COS или IOS для устройства. Для просмотра образа используется команда `dir`, за которой указывается имя устройства, например, `dir bootflash:` или `dir slot0:`.

б) Загрузка образа с flash-карты.

```
rommon> boot [device:][:][image]
```

Для того чтобы загрузиться из режима ROM-монитор, используется команда `boot`. Если команда вводится без имени устройства или файла, то используется поле BOOT конфигурационных переменных. Если это поле пустое или введено неверное имя файла, пользователь возвращается к приглашению `rommon>`. В случае, когда при использовании команды `boot` указывается имя файла, значение загрузочной переменной игнорируется и используется указанный файл.

Внимание!

Загрузочные переменные и имена файлов чувствительны к регистру символов. Если имя файла указано неверно, в имени пропущен символ или используется неверный регистр, то файл не будет найден, и коммутатор вернет пользователя в режим `rommon>`. Рекомендуется просмотреть содержимое flash-устройства, выделить и скопировать имя файла в буфер с помощью команд группы Edit в терминальном приложении.

5. Передача данных по протоколу Xmodem.

```
rommon> xmodem
```

Данная команда инициирует средства протокола Xmodem для ROM монитор. Используя ее, можно загрузить коммутатор с использованием файла, который

размещен на персональном компьютере, подключенном к консольному порту. Для того чтобы начать асинхронную передачу с использованием протокола Xmodem и отправить файл с жесткого диска компьютера во flash-устройство ROM-монитора, используется терминальное программное обеспечение компьютера. После того как коммутатор загрузится с использованием в качестве источника программного образа компьютера, операционная система активизируется и появится возможность скопировать необходимый файл во flash-память. Этот процесс может занять много времени, поэтому такую загрузку необходимо рассматривать в качестве последней возможности при восстановлении утраченного или поврежденного образа операционной системы.

В этой главе...

- **2.1: семейство коммутаторов Catalyst.** В этом разделе представлено краткое описание платформ Cisco Catalyst, их возможностей, а также поддерживаемые ими операционные системы.
- **2.2: конструкции коммутируемых территориальных сетей.** В этом разделе представлен краткий перечень правил и соображений, которые могут использоваться при проектировании коммутируемой сети масштаба предприятия.

Глава 2

Функции коммутатора

2.1: семейство коммутаторов Catalyst

Линейка коммутаторов Catalyst постоянно расширяется. Начиная с приобретения компаний Grand Junction, Kalpana и Crescendo в середине девяностых годов двадцатого века и слияния с компанией Granite в 1999 году, корпорации Cisco Systems собрала вместе несколько лучших в мире инженеров, занимающихся проблемами коммутации. В результате их совместной работы появились семейство коммутирующих продуктов, которые предоставляют различные функции для внедрения в территориальных сетях.

Одной из главных трудностей при выборе и внедрении коммутатора в сеть является понимание того, какие функции осуществляются данным устройством, а также как коммутатор функционирует внутри сетевой конструкции. Целью раздела является предоставление читателю краткого обзора современных платформ коммутаторов Catalyst и их основных функций. В дополнение к этому в одном из подразделов кратко описываются операционные системы, поддерживаемые различными платформами.

Коммутаторы Catalyst серий 3500XL и 2900MXL

Коммутаторы Catalyst серий 3500XL и 2900MXL обеспечивают базовую связь для пользовательских устройств. Эти коммутаторы называются *коммутаторами доступа к сети (access switches)* и имеют различную плотность портов: от 12 до 48. Коммутаторы серии 3500XL поддерживают разъемы гигабитных интерфейсов для высоких гигабитовых каналов. Коммутаторы серии 2900MXL представляют собой модульные устройства, поддерживающие различные модульные соединения, включая модуль *клиента LAN-эмуляции ATM (LAN emulation client — LEC)* для подключения к ATM LANE-сети. Эти коммутаторы предоставляют приемлемые по стоимости решения для доступа в коммутируемую среду, а также функции передачи второго уровня с такими службами, как ограниченная классификация и механизм планирования пересылки для реализации средств *качества обслуживания (Quality of Service — QoS)*, магистральные соединения и возможность создания FiberChannel-каналов. Кроме того, для облегчения процесса администрирования несколько коммутаторов этой серии можно объединить в кластер.

Коммутаторы Catalyst серии 2950

Устройство Catalyst серии 2950 является коммутатором, который, как правило, обеспечивает доступ к сети конечных пользователей, а также предоставляет расширенные функции. Коммутаторы этой серии выпускаются с различной плотностью портов и некоторыми отличиями в скоростях передачи портов и разъемов, характерных для передающей среды. Один из продуктов этой серии обеспечивает связь на уровне 10/100/1000 Мбит/с с использованием медных кабелей. Данный коммутатор допускает развертывание гигабитного канала к уровню доступа сети. Коммутаторы серии 2950 обеспечивают перенаправление пакетов второго уровня, а также обладают множеством таких же возможностей, что и коммутаторы Catalyst серии XL: возможность установки магистрального соединения и создания EtherChannel каналов. Кроме того, коммутаторы этой серии добавляют функции безопасности третьего и четвертого уровней с *использованием списков доступа (Access Control Lists — ACL)* для VLAN-сетей, а также улучшенную классификацию и планирование качества обслуживания, основанные на информации третьего и четвертого уровней. Коммутаторы серии 2950 представляют собой следующее поколение коммутаторов доступа к территориальным сети.

Коммутаторы Catalyst серии 3550

Устройства Catalyst серии 3550 — коммутаторы среднего класса, которые в зависимости от программного обеспечения способны предоставлять службы второго уровня либо службы второго и третьего уровней в одном устройстве. Коммутаторы этой серии поставляются с различной плотностью портов и обеспечивают поддержку технологий Fast Ethernet и Gigabit Ethernet. В данном коммутаторе имеются порты, которые могут быть непосредственно сконфигурированы в качестве интерфейсов третьего уровня. Для коммутации третьего уровня в коммутаторе также могут использоваться виртуальные (VLAN) интерфейсы. В дополнение к контролю доступа для портов второго или третьего уровней коммутатор поддерживает функции второго уровня на портовой основе или базовой связи второго уровня и такие улучшенные функции, как магистральное соединение, объединение каналов (channeling), классификация и маркирование службы QoS.

Благодаря гибкости коммутаторов серии 3550 эти устройства превосходно работают в небольших и среднего размера территориальных средах, а также могут внедряться в качестве коммутаторов доступа к сети или распределения. Коммутаторы этой серии как устройства среднего класса, которые работают на уровне скорости передачи по проводнику, заменили коммутаторы модели 2948G-L3. В устройстве Catalyst 3550 может использоваться либо *стандартный многоуровневый образ программного обеспечения (Standard Multilayer software Image — SMI)* для механизмов коммутации второго уровня, либо *усовершенствованный многоуровневый образ программного обеспечения (Enhanced Multilayer software Image — EMI)* для использования комбинированных средств коммутации второго и третьего уровней.

Коммутаторы Catalyst серии 5000

Коммутаторы серии 5000 являются одними из основных коммутирующих элементов в территориальных сетях Cisco. Данный коммутатор — модульный продукт, предоставляющий как поддержку различных передающих сред, так и различную плотность портов. В серию включаются продукты серий 5000 и 5500. Коммутаторы серии 5500 использовались на

многих сетях как коммутаторы уровня распределения и магистральные коммутаторы, однако в настоящее время ввиду ограничений, характерных для их объединительной платы, применяются в качестве коммутаторов уровней доступа к сети и распределения. Коммутаторы этой серии в дополнение к модулям маршрутизации и WAN-модулям поддерживают модули ATM, FDDI, Token Ring, Ethernet, Fast Ethernet и Gigabit Ethernet. Продукты серии 5500 предоставляют расширенные функции, такие, как резервирование модулей Supervisor и возможность многоуровневой коммутации. Эти коммутаторы также предоставляют основную QoS-классификацию и службы расписания.

Коммутаторы Catalyst серии 4000

Устройство Catalyst 4000 — коммутатор среднего класса, способный функционировать в качестве коммутатора уровня доступа или распределения с высокой плотностью портов, а также устройства магистрали сети с низкой плотностью портов. Устройства этой серии также являются модульными коммутаторами, обеспечивающими службы второго и третьего уровней. В коммутаторе имеется поддержка Ethernet-технологий и модуль третьего уровня. Блок Supervisor III для коммутаторов серии 4000 содержит интегрированный модуль третьего уровня и может работать под управлением программного обеспечения Cisco IOS. Эти коммутаторы также осуществляют магистральные функции второго уровня и обеспечивают поддержку каналов EtherChannel. Устройство Catalyst 4000 предоставляет некоторые основные функции QoS-классификации и планирования.

Коммутаторы Catalyst серии 6000

Серия 6000 является флагманом линейки продуктов Catalyst. Коммутатор этой серии наиболее надежен, обладает наилучшей поддержкой объединительной платы и является наиболее гибким из всех продуктов линейки Catalyst. Модульный коммутатор способен функционировать в качестве коммутатора уровня доступа с высокой плотностью портов, коммутатора распределения второго/третьего уровней, а также основного магистрального коммутатора второго или третьего уровня, работающего на скорости среды передачи данных. В дополнение к высокоскоростным возможностям Ethernet-коммутации в данном устройстве предоставляются различные опции для поддержки расширенных функций, таких, как голосовые службы, коммутация по содержимому (content switching), обнаружение вторжений (intrusion detection), функции анализа сети, оптические службы, 10 Гбит/с Ethernet, поддержка брандмауэров и службы шифрования. Все функции осуществляются на уровне скорости передачи данных по проводнику. В дополнение к описанным выше службам шасси 6500 поддерживает связь модуля коммутационной матрицы (fabric module) для обеспечения взаимного подключения плат вместо использования объединительной платы с пропускной способностью 32 Гбит/с. С таким модулем коммутатор 6509 или 6513, полностью заполненный платами, подключаемыми к коммутационной матрице (fabric-enabled cards), обладает общей скоростью передачи в 256 Гбит/с. Этот коммутатор также предоставляет поддержку резервирования и параметры высокой надежности. Коммутаторы серии 6000 продолжают развиваться как новые продукты, обеспечивающие большую гибкость и функциональность.

Внимание!

Несмотря на то что корпорация Cisco объявила 15 мая 2002 года датой окончания продажи шасси серий 6006 и 6009, важно отметить, что единственным отличием шасси серий 6000 и

6500 имеются разъемы для плат коммутационной матрицы. Шасси без соответствующих плат и модулей не предоставляют усовершенствованных служб. Таким образом, коммутатор серии 6500 без модулей и плат коммутационной матрицы функционирует так же, как и коммутатор Catalyst серии 6000. В текущей главе при описании упомянутых продуктов семейство «коммутаторов» обозначается как серия 6000.

Операционные системы коммутаторов

Поскольку многообразие существующих продуктов Catalyst является прямым результатом приобретения других компаний, слияния с ними и продолжающегося развития, нетрудно поверить, что существует также и множество операционных систем (Operating System — OS), позволяющих конфигурировать данные устройства и управлять ими. Главной трудностью при работе с коммутаторами Cisco является понимание различий между операционными системами и различий в методах конфигурирования одних и тех же функций на различных платформах. Основной целью данной книги является определение необходимых этапов конфигурирования функции и описание различий в таких этапах для разных операционных систем. Прежде чем настраивать операционные системы, важно научиться различать их работу на различных платформах.

Различные серии коммутаторов Cisco используют две основные операционные системы. Первая — *Cisco Internetwork Operating System*, или просто *Cisco IOS* (*межсетевая операционная система Cisco*). Она основывается на том же ядре и оболочке, которые используются для маршрутизаторов. Для коммутаторов существуют три производные этой OS.

- **Операционная система IOS второго уровня (Layer 2 IOS).** Система Cisco IOS устанавливается на устройстве, передающем пакеты только на основании информации второго уровня. Устройствами, использующими IOS второго уровня, являются коммутаторы моделей 3500XL и 2950.
- **Операционная система IOS третьего уровня (Layer 3 IOS).** Система IOS третьего уровня запускается на коммутаторе, осуществляющем передачу пакетов третьего уровня для своих интерфейсов. Такая же система используется на маршрутизаторах. Однако в среде коммутаторов она встречается в продуктах 2948G-L3, модуле коммутации маршрутов (Route Switch Module — RSM), модуле многоуровневой коммутации (Multilayer Switch Module — MSM), функциональной плате коммутации маршрутов (Route Switch Feature Card — RSFC), функциональной плате многоуровневого коммутатора (Multilayer Switch Feature Card — MSFC).
- **Операционная система IOS второго/третьего уровней (Layer 2/Layer 3 IOS).** Данная система IOS запускается на устройстве, в котором может существовать порт, функционирующий подобно порту маршрутизатора (устройства третьего уровня) или подобно порту коммутатора (устройства второго уровня) в зависимости от конфигурации этого устройства. Данную операционную систему иногда называют интегрированной системой IOS *(Integrated IOS)*, поскольку в ней объединяются функции обоих уровней эталонной модели *взаимодействия открытых систем (Open System Integration — OSI)*. Система работает на коммутаторе модели 3550. Кроме того, она может работать на коммутаторе серии 4000 с установленной платой Supervisor III или коммутаторе серии 6000 с MSFC-платой. Если код выполняется на одной из этих платформ, то система также называется *собственной системой* IOS, или *Supervisor* IOS.

Структура команд, этапы конфигурирования и управляющие системой отличаются в упомянутых вариантах IOS. Кроме того, указанные три системы несколько отличаются и поддержке в некоторых конфигурационных параметрах. В книге указаны различия между ними, где это необходимо.

Второй операционной системой для семейства коммутаторов Catalyst является COS (*Catalyst Operating System* – *операционная система Catalyst*, которую иногда называют *CatOS*. Эта система предоставляет только поддержку второго уровня и работает на коммутаторах серии 4000, 5000 и 6000. Благодаря простоте использования командной структуры данной операционной системы, она является весьма популярной при обеспечении коммутации второго уровня. В табл. 2.1 представлены различные операционные системы и поддерживающие их платформы.

Таблица 2.1. Операционные системы и поддерживающие их платформы

Операционная система	Cisco IOS			Операционная система Catalyst
Поддержка OSI-уровней	Только второй уровень	Только третий уровень	Второй и третий уровни	Только второй уровень
Платформа	2900MXL, 3500XL, 2950	2948G-L3, 4908G-L3, RSM/MSM, RSFC/MSFC, все маршрутизаторы, модуль служб третьего уровня	3550, серия 4000[1], серия 6000[2]	Серии 5000, 4000, 6000

Для проектирования и конфигурирования территориальных сетей на основе продуктов корпорации Cisco используются различные коммутаторы и операционные системы. При чтении этой книги следует учитывать используемую операционную систему и платформу.

2.2: конструкции коммутируемых территориальных сетей

При проектировании коммутируемой сети необходимо учитывать множество факторов. Расширение или перепроектирование крупной сети предприятия или территориальной сети может оказаться сложным громоздким делом. Существует общепринятый, организованный подход к проектированию коммутируемой сети, который способен упростить процесс проектирования, а также сделать сеть более эффективной и расширяемой.

Этот раздел скомпонован в виде краткого справочного "перечня" методик, примерных правил, а также идей, способствующих полному осмыслению общей структуры и конфи-

[1] *Коммутатор Catalyst серии 4000 для работы с интегрированной (Supervisor) IOS требует наличия модуля Supervisor III.*

[2] *Коммутатор Catalyst серии 6000 для работы с интегрированной (Supervisor) IOS требует наличия модуля Supervisor с платой MSFC*

гурации сети. Многие пункты "перечня" включают в себя ссылки на соответствующие разделы этой книги, в которых рассматривается функции коммутаторов.

1. Сегментация локальных сетей на небольшие коллизионные домены, возможна при помощи LAN-коммутаторов.
2. Организация сети предприятия в иерархическую структуру.

   Сеть, спроектированная на основе многоуровневой структуры, предоставляет базу для предсказуемой работы, последовательных задержек (с учетом количества транзитных коммутаторов) из любой точки сети, а также расширяемость сети. Если потребуется расширить сеть, то в существующую структуру можно добавить большое количество блоков коммутаторов.

   На рис. 2.1 представлена базовая иерархия сети, разделенная на три обособленных уровня.

   - **Уровень доступа** (access layer) состоит из коммутаторов, подключенных к конечным пользователям.
   - **Уровень распределения** (distribution layer) состоит из коммутаторов, суммирующих трафик, поступающий из уровня доступа.
   - **Основной уровень** (core layer) состоит из коммутаторов, суммирующих трафик, поступающий с уровней распределения.

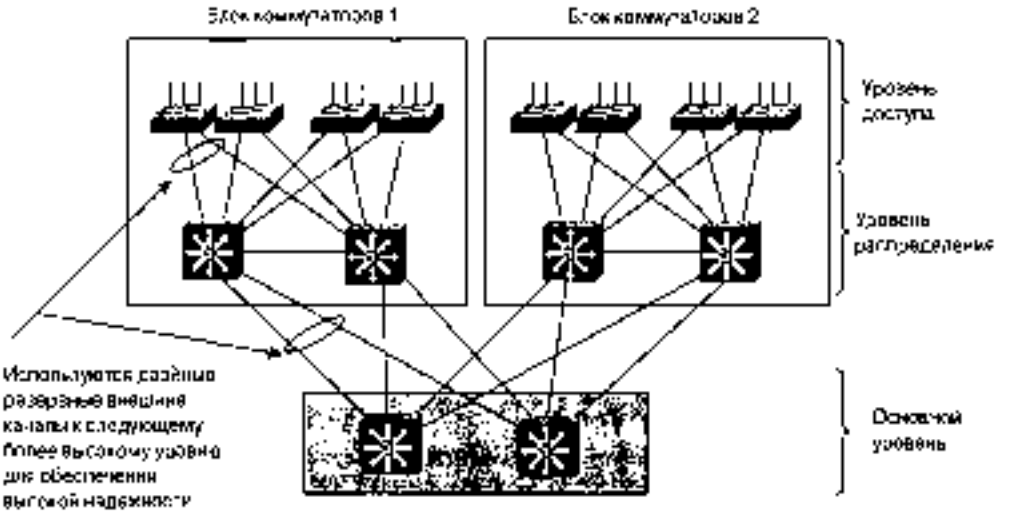


Рис. 2.1. Уровни иерархической конструкции сети

Совет

В сетях предприятий небольшого и среднего размера уровень распределения можно исключить. В таком случае коммутаторы уровня доступа непосредственно подключаются к основному уровню. Такая схема называется конструкцией с вырожденной базовой сетью (collapsed core).

Чтобы обеспечить высокую надежность, каждый коммутатор на сетевом уровне должен иметь удвоенные или резервные внешние каналы к двум коммутаторам, распо-

ложенным на следующем, более высоком уровне. В случае выхода из строя канала или отказа всего коммутатора можно быстро задействовать дополнительный внешний канал. Операции по восстановлению внешнего канала в случае сбоев на втором уровне поддерживаются *протоколом распределенного связующего дерева (Spanning Tree Protocol — STP)* или протоколами маршрутизации на третьем уровне.

3. Реализация функции коммутации на каждом уровне иерархии.
   - **Уровень доступа.** Коммутаторы этого уровня, как правило, обладают высокой плотностью портов, низкой стоимостью и имеют функции, обеспечивающие доступ пользователей или безопасность, а также несколько высокоскоростных портов для внешних каналов. Обычно коммутации второго уровня достаточно, однако коммутация третьего уровня способна обеспечить более высокую надежность для таких приложений, как IP-телефония.
   - **Уровень распределения.** Коммутаторы этого уровня обладают плотностью высокоскоростных портов и обеспечивают более высокую производительность коммутации в идеальном случае на третьем уровне.
   - **Основной уровень.** Этот уровень должен основываться на коммутаторах, обладающих наивысшей производительностью в сети и суммирующих трафик от коммутаторов уровня распределения. На основном уровне могут эффективно использоваться коммутаторы второго уровня, хотя коммутация на третьем уровне повышает надежность и добавляет улучшенные функции QoS. Обычно для поддержки основного уровня всего предприятия вам будет двух коммутаторов достаточно.
4. Определение требуемых ресурсов в сети, которые предоставляют общие функции. Данные ресурсы станут модулями или "строительными блоками" конструкции сети. На рис. 2.2 представлены несколько примеров таких блоков, а также их расположение в сетевой иерархии.

Совет

Сеть, изображенная на рис. 2.2, для простоты представлена с одинарными внешними каналами к более высоким уровням. В реальной сети для обеспечения более высокой надежности следует всегда добавлять либо удвоенные, либо резервные внешние каналы к коммутаторам на верхнем уровне иерархии.

В данном случае каждый коммутатор уровня доступа обладал бы двумя внешними каналами к ближайшим коммутаторам уровня распределения. В дополнение к этому каждый коммутатор уровня распределения в каждом блоке диаграммы имел бы два внешних канала к двум коммутаторам основного уровня. Иными словами, основные принципы, проиллюстрированные на рис. 2.1, следует применить к планировке сети предприятия на рис. 2.2.

- **Группы серверов и мейнфреймов** — *блоки серверов (server blocks)* и *блоки мейнфреймов (mainframe blocks)* соответственно.
- **Доступ к сети Internet, электронная коммерция или серверные группы внешних сетей, а также группы брандмауэров** — этот блок называется *Internet-блоком*.
- **Удаленный доступ** — *WAN-блок*.
- **Телефонные службы и шлюзы** — *PSTN-блок*.

- **Устаревшие сети (Token Ring, FDDI и другие)** — блок, весьма сходный с WAN-блоком, использующий маршрутизатор для обеспечения связи с сетями на основе различных передающих сред и технологий.
- **Общие рабочие группы пользователей** — конечные пользователи, находящиеся в одном здании, на одном этаже или на одном участке этажа, входят в состав блока *коммутаторов*. В блоке коммутаторов обычно группируются коммутаторы уровня доступа и коммутаторы распределения, к которым они подключены.

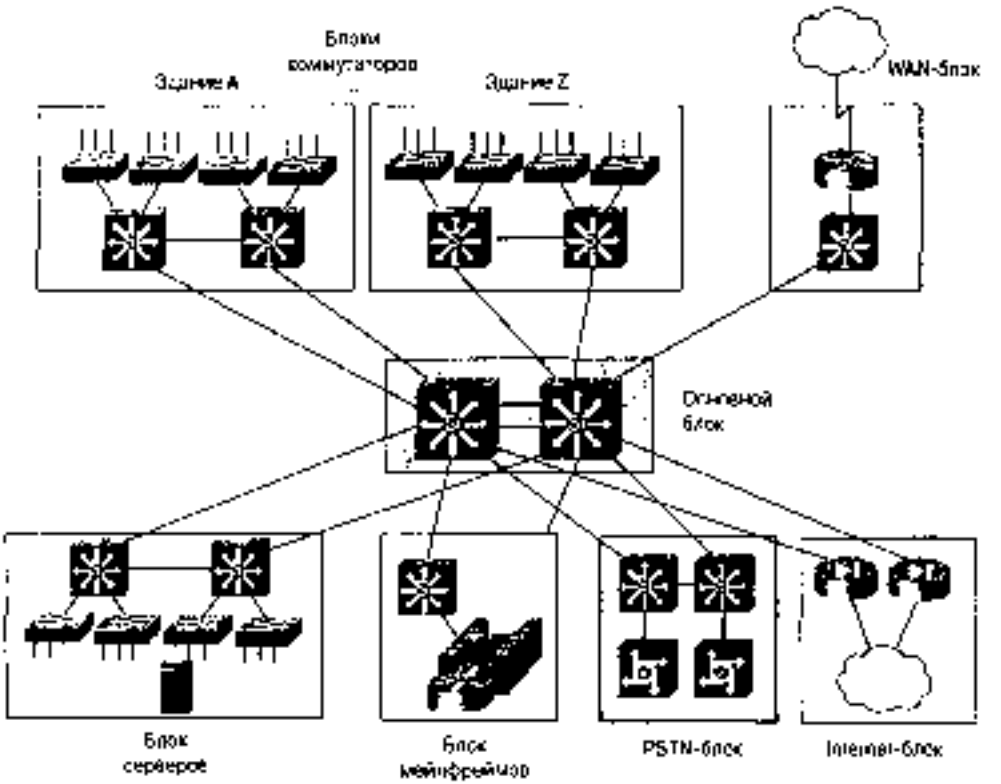


Рис. 2.2. Модульный подход к проектированию территориальной сети

5. Определение функций обеспечения высокой надежности и резервирования, которые могут использоваться в каждом строительном блоке сети.

   а) Основной блок.

   - Если используются коммутаторы второго уровня, не следует создавать петли для распределенного связующего дерева путем соединения двух коммутаторов основного уровня.
   - Для каждой VLAN-сети необходимо определить и сконфигурировать коммутаторы как основного, так и запасного корневых мостов. Как правило, корневой мост располагают вблизи основного уровня. За более подробной информацией по этому вопросу обратитесь к разделу "7.2. конфигурирование протокола STP".

- Если используются коммутаторы третьего уровня, то коммутаторы основного уровня следует подключать к нескольким каналам. За более подробной информацией по этому вопросу обратитесь к разделу "4.4: порты EtherChannel".
- В основе третьего уровня для обеспечения резервирования адресов шлюза необходимо использовать протокол HSRP (Hot Standby Router Protocol — протокол резервной маршрутизации). За более подробной информацией по этому вопросу обратитесь к разделу "8.6: резервирование маршрутизаторов с помощью протокола HSRP".
- Каждый коммутатор основного уровня для создания полного резервирования следует подключить ко всем коммутаторам уровня распределения. Если в основном уровне или уровне распределения не используются коммутаторы третьего уровня, то для сокращения времени сходимости протокола STP следует использовать функцию BackboneFast STP. За более подробной информацией по этому вопросу обратитесь к разделу "7.2: конфигурирование протокола STP".

б) Блок серверов.

- Использование резервных восходящих каналов к уровню распределения или основному уровню. Для ускоренного восстановления после сбоев следует использовать функцию STP UplinkFast (за более подробной информацией по этому вопросу обратитесь к разделу "7.2: конфигурирование протокола STP") или протокол HSRP (раздел "8.6: резервирование маршрутизаторов с помощью протокола HSRP").
- Следует рассмотреть использование сдвоенных *сетевых интерфейсных плат (Network Interface Card — NIC)* в серверах для обеспечения резервирования. Сетевые платы подключаются к различным платам или модулям коммутаторов.

в) Internet-блок.

- Для распределения трафика среди множества серверов в группе используется балансировка нагрузки на серверы (Server Load Balancing). За более подробной информацией по этому вопросу обратитесь к разделу "10.1: технология SLB".
- Для распределения трафика среди множества брандмауэров в группе используется балансировка нагрузки на брандмауэрах (Firewall Load Balancing). За более подробной информацией по этому вопросу обратитесь к разделу "10.2: балансировка нагрузки на брандмауэры".

г) Блоки коммутаторов.

- Каждый коммутатор уровня доступа обладает двойными избыточными каналами к двум отдельным коммутаторам уровня распределения.
- С целью сокращения времени восстановления после сбоев на коммутаторах уровня доступа используется функция UplinkFast протокола STP.
- Для сокращения времени первоначального подключения пользователей на портах уровня доступа используется функция PortFast протокола STP.
- Для распределения нагрузки между избыточными каналами уровня доступа следует настроить параметры протокола STP так, чтобы одна VLAN-сеть

доступа занимала один внешний канал, а вторая VLAN-сеть — другой (второго уровня на уровне распределения). В противном случае следует настроить HSRP-приоритеты в устройствах распределения третьего уровня так, чтобы один коммутатор уровня распределения поддерживал одну VLAN-сеть доступа, а второй — другую.

- Если на уровне распределения используются устройства третьего уровня, то нужно применить команду passive для интерфейсов к уровню доступа, где нет других маршрутизаторов.

б. Другие соображения.

а) Для каждой VLAN-сети следует настроить корневой STP-мост и резервный корневой мост как можно ближе к основному уровню. За более подробной информацией по этому вопросу обратитесь к разделу "7.2. конфигурирование протокола STP".

б) Широковещательные домены.

- Следует ограничить размер широковещательных доменов путем управления размером VLAN-сетей. Допускается расширять VLAN-сети на любом участке сети, однако за таким расширением последует расширение границ широковещательного трафика.
- Следует рассмотреть использование функций подавления широковещания на портах коммутатора. За более подробной информацией по этому вопросу обратитесь к разделу "11.1. подавление широковещания".

в) *Протокол магистральных каналов VLAN-сетей (VLAN Trunking Protocol — VTP).*

- Конфигурирование ближайших к основному уровню VTP серверов.
- Использование VTP-отсекания или ручное конфигурирование определенных VLAN-сетей для передачи данных по магистральным каналам. Эта методика сокращает нежелательный широковещательный трафик в магистральных каналах.

г) Масштабирование магистральных каналов.

- Объединение нескольких магистральных каналов в EtherChannel. Для обеспечения отказоустойчивости следует разделить EtherChannel-канал между модулями коммутатора. За более подробной информацией по этому вопросу обратитесь к разделу "4.4 порты EtherChannel".
- Не следует конфигурировать согласование магистральных каналов. Рекомендуется использовать режим "on" (включен). За более подробной информацией по этому вопросу обратитесь к разделу "6.3 транкинг".

д) Качество обслуживания (QoS).

- Конфигурирование параметров QoS на каждом коммутаторе в сети. Необходимо обеспечить соответствующую поддержку качества обслуживания в сквозном режиме. За более подробной информацией по этому вопросу обратитесь к разделу "13.2. Конфигурирование средств QoS".
- Расширение доверительных границ QoS (QoS trust boundary) до граничных устройств (например, IP-телефонов), которые могут обеспечить соответствующие функции.
- Использование упорядочения потоков данных некритичных приложений.

е) Резервные модули коммутатора.

- Следует принять во внимание использование резервных блоков Supervisor в коммутаторах серверной группы, где узлы имеют одно подключение (одну сетевую плату).
- Если резервные внешние каналы присутствуют на каждом сетевом уровне, то два физически обособленных коммутатора всегда будут обеспечивать отказоустойчивость. Рекомендуется использовать запасные блоки Supervisor в коммутаторах основного уровня или уровня распределения, где доступны только одиночные внешние каналы.
- Использование высоконадежной связи между блоками Supervisor в шасси. Необходимо включить контроль версий так, чтобы существовала возможность обновить операционную систему без простоя коммутатора. За более подробной информацией по этому вопросу обратитесь к разделу "2.6: резервные блоки Supervisor".

ж) Защита портов и аутентификация

- Существует возможность контролировать MAC-адреса конечных пользователей или количество пользователей, подключенных к порту коммутатора уровня доступа с функциями защиты портов (port security). За более подробной информацией по этому вопросу обратитесь к разделу "11.3: функция обеспечения безопасности портов".
- Можно также включить аутентификацию пользователей на портах коммутаторов уровня доступа с использованием спецификации 802.1x. За более подробной информацией по этому вопросу обратитесь к разделу "11.5: аутентификация на коммутаторе".
- Контроль доступа к VLAN-сетям осуществляется с помощью VLAN ACL-списков. За более подробной информацией по этому вопросу обратитесь к разделу "11.4: списки доступа VLAN-сетей".

Дополнительная литература

Рекомендуемые ниже источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Семейства коммутаторов Catalyst

Краткое справочное руководство по продуктам Cisco / Cisco Product Quick Reference Guide — CPQRG: www.cisco.com/warp/customer/752/qrg/. В частности, по коммутаторам Catalyst существует отдельный раздел — www.cisco.com/warp/customer/752/qrg/cpqrg2.htm.

Справочник по программному обеспечению (Software Advisor) — средство для сравнения и сопоставления функции операционных систем COS и IOS, их версий и поддерживающих аппаратных платформ: www.cisco.com/cgi-bin/Support/CompNav/Index.pl. Для доступа к нему требуется регистрационное имя в системе CCO1.

Официальный справочник по операционной системе IOS

www.cisco.com/warp/public/cc/pd/si/casi/ca6000/tech/car65_wp.htm

Конструкции коммутируемых территориальных сетей

Проектирование кампусных территориальных сетей: принципы и структура (Gigabit Campus Network Design — Principles and Architecture): www.cisco.com/warp/public/cc/so/neso/lnso/cpso/gcnd_wp.htm.

Проектирование кампусных территориальных сетей: конфигурация и анализ средств восстановления после сбоев (Gigabit Campus Design — Configuration and Recovery Analysis): www.cisco.com/warp/public/cc/so/neso/lnso/cpso/canp_wp.htm.

Проектирование высокопроизводительных территориальных intranet-сетей с многоуровневой коммутацией (Designing High-Performance Campus Intranets with Multilayer Switching): www.cisco.com/warp/public/cc/so/cuso/epso/entdes/highd_wp.htm.

Matthew Birkner, *Cisco Internetwork Design*, Cisco Press.

Karen Webb, *Building Cisco Multilayer Switched Networks*, Cisco Press.

Tim Boyles and David Hucaby, *CCNP Switching Exam Certification Guide*, Cisco Press.

В этой главе...

- **3.1: приглашения командной строки и системные заставки.** В разделе описывается метод настройки приглашений и заставок для идентификации коммутатора.
- **3.2: IP-адресация и службы.** В разделе объясняется методика конфигурирования IP-адресации и служб для управления коммутатором с помощью протокола TCP/IP.
- **3.3: пароли и их восстановление.** В этом разделе описывается процесс установки паролей коммутаторов, а также методы восстановления потерянных и неизвестных паролей.
- **3.4: управление модулями.** В этом разделе описывается контроль мощности модуля, перезагрузка модуля и управление конфигурацией отдельных блоков модульных коммутаторов.
- **3.5: управление файлами и параметры загрузки.** В разделе поясняется процесс управления конфигурацией и системными файлами, а также управление процессом загрузки.
- **3.6: резервные блоки Supervisor.** В разделе описана функция, с помощью которой синхронизируется конфигурация резервных модулей Supervisor, а также методика управления этой функцией.
- **3.7: протокол обнаружения устройств Cisco.** В этом разделе описано взаимодействие *протокола обнаружения устройств Cisco (Cisco Discovery Protocol — CDP)* с устройствами производства корпорации Cisco, а также управление CDP-функциями для портов коммутатора.
- **3.8: установка времени и даты.** В разделе представлены основные этапы, необходимые для конфигурирования даты и времени в коммутаторе путем установки этих параметров вручную, а также с помощью *синхронизирующего сетевого протокола (Network Time Protocol — NTP).*

Глава 3

Конфигурирование блока Supervisor

3.1: приглашения командной строки и системные заставки

- Приглашение коммутатора предоставляет пользователю информативное имя в каждой точке ввода командной строки, помогая таким образом пользователям идентифицировать устройство, которое они администрируют
- В большинстве операционных систем приглашение следует за именем системы или узла, однако эту настройку можно изменить.
- Системные заставки позволяют идентифицировать коммутаторы, а также предоставляют информацию о политике безопасности и процедурах мониторинга.
- Конфигурирование приглашения и заставок является необязательным

Конфигурация приглашения

1. Конфигурирование приглашения *(необязательно)*.

   a) Конфигурирование приглашения путем назначения имени устройству.

Система COS	`set system name string`
Система IOS	`hostname string` (режим глобальной конфигурации)

Стандартным именем узла для IOS устройства является "Switch" (коммутатор) или "Router" (маршрутизатор) в зависимости от функции (второго или третьего уровня), выполняемой этим коммутатором. COS коммутатора не имеет имени, поэтому стандартным приглашением является слово "console" (консоль).

Внимание!

В ранних версиях операционной системы Catalyst (COS 3.2 и ниже) установка системного имени не влияла на приглашение. Для того чтобы настроить приглашение в устройствах, использующих указанные версии COS, следует придерживаться рекомендаций пункта б) ниже.

б) Явная настройка приглашения.

Система COS	`set system prompt string`
Система IOS	`prompt string` (режим глобальной конфигурации)

COS-коммутаторы не имеют имени, поэтому стандартным приглашением является слово "console" (консоль). Для IOS-устройства стандартным именем узла является "Switch" (коммутатор) или "Router" (маршрутизатор) в зависимости от выполняемой функции.

Конфигурирование системной заставки

1. Конфигурирование заставки *сообщения дня (Message of the Day — MOTD)* (необязательно).

   MOTD-заставки не являются необходимыми для функционирования системы, однако они чрезвычайно полезны для идентификации политики безопасности, относящейся к доступу к сетевому устройству. Для COS-устройств системные заставки являются способом идентификации коммутатора без регистрации в его системе.

   а) Конфигурирование MOTD-заставки.

Система COS	`set banner motd & string &`
Система IOS	`banner motd & string &` (режим глобальной конфигурации)

Текст заставки печатается между ограничивающими символами (в примере выше — амперсанд [&]). Ограничивающий символ печатается в начале и конце заставки, которая может включать в себя несколько строк, разрывов строки и слов. Ограничивающим символом может быть любой символ, не являющийся частью текста системной заставки.

Внимание!

Системные заставки ограничиваются в размерах устройством и операционной системой. Согласованного числа символов или ограничений не существует. Ранние серии коммутаторов 1900 и 2800 имели ограничение равное 400 символам, или 20 строкам. В последней версии системы COS максимальный размер сообщения составляет 3070 символов.

Пример конфигурирования функции

В качестве иллюстрации продемонстрируем типичные настройки для системного имени, приглашения и заставки.

Ниже приведен пример конфигурации для операционной системы Catalyst

```
Console (enable)>set system name Core_Switch1
Core_Switch1 (enable)>set banner motd *
This is Core_Switch1 for the XYZ corporation.
You have accessed a restricted device, unauthorized logins are prohibited.
*
Core_Switch1 (enable)>
```

Ниже приведен пример соответствующей конфигурации для Supervisor IOS.

```
Switch(config)#hostname Core_Switch1
Core_Switch1(config)#banner motd *
This is Core_Switch1 for the XYZ corporation.
You have accessed a restricted device, unauthorized logins are
prohibited.
*
Core_Switch1(config)#end
Core_Switch1#copy running-config startup-config
```

3.2: IP-адресация и службы

- В коммутаторах IP-адреса и службы используются в административных целях.
- IP-адреса могут быть установлены или получены с помощью *протокола динамического конфигурирования узла (Dynamic Host Configuration Protocol — DHCP)*, *протокола начальной загрузки (BOOTstrap Protocol — BOOTP)* или *протокола обратного преобразования адресов (Reverse Address Resolution Protocol — RARP)*.
- Чтобы обеспечить обмен данными с устройствами, которые не являются локальными для управляемой сети, устанавливаются шлюзы, маршруты к сетям и стандартные маршруты.
- Для преобразования имен компьютеров могут использоваться статические записи или DNS-серверы.
- В некоторых коммутаторах доступны службы *протокола передачи гипертекста (Hypertext Transfer Protocol — HTTP)*, обеспечивающие интерфейс конфигурирования данных устройств.
- *Простой протокол управления сетью (Simple Network Management Protocol — SNMP)* также позволяет конфигурировать и администрировать коммутаторы.

Конфигурирование административного IP-адреса

IP-адреса используются в коммутаторах второго уровня только в целях администрирования. Данный этап не является обязательным для функционирования коммутатора. В случае, если IP-адрес не был задан, единственным способом управления коммутатором является консольное соединение.

1. Конфигурирование IP-адреса (*необязательный этап, но рекомендуется его выполнить*).

   а) Конфигурирование IP-адреса вручную.

Система COS	`set interface sc0` *[vlannumber] address mask*
Система IOS	`interface vlan` *vlannumber* (режим глобальной конфигурации) `ip address` *address mask* (режим конфигурирования интерфейса или подынтерфейса) `management` (режим конфигурирования интерфейса или подынтерфейса)

Коммутаторы Catalyst могут иметь активный административный адрес (management address) только в одной VLAN-сети. В IOS-коммутаторах второго

уровня активная VLAN-сеть определяется с помощью административной команды. Сеть VLAN 1 является стандартной административной сетью как для IOS-, так и для COS-коммутаторов. В IOS-коммутаторах второго уровня, если сеть VLAN 1 не является административной сетью, приглашение использует "подинтерфейс"

На COS-коммутаторах, в которых используется операционная система COS 5.x или выше, также может быть настроена маска в виде битового или CIDR-представления (*CIDR, Classless InterDomain Routing — бесклассовая междоменная маршрутизация*). Например, `set interface sc0 10.1.1.1/24`

Для того чтобы просмотреть IP-конфигурацию, в COS-устройствах используется команда **show interface**, а в IOS-устройствах — команда **show interface vlan** *n* (где *n* — номер данной VLAN-сети).

Внимание!

В этом разделе при изучении адресации рассматриваются только административные адреса и интерфейсы второго уровня. Интерфейсы третьего уровня обсуждаются в главе 5, "Конфигурирование интерфейсов третьего уровня".

б) Автоматическое получение IP-адреса *(не рекомендуется)*.

Коммутатор может запрашивать адрес у таких служб, как RARP, BOOTP и DHCP. Такой подход использовать не рекомендуется, поскольку не исключено, что адрес мог измениться из-за протокола DHCP, если время адреса не является постоянной или статической (т.е. срок действия аренды никогда не истекает или для MAC-адреса коммутатора зарезервирован определенный IP-адрес). Также этот метод получения адреса означает, что изменение аппаратного обеспечения может повлечь за собой проблемы с BOOTP- или статическим DHCP-адресом. Кроме того, указанные службы поддерживаются не всеми коммутаторами. COS-устройство в том случае, если адрес не сконфигурирован (что является стандартной установкой), запрашивает RARP- или BOOTP/DHCP-адрес после загрузки. Если требуется удалить адрес и настроить COS-коммутатор на автоматическое преобразование IP-адреса, то необходимо установить адрес 0.0.0.0, а затем перезапустить коммутатор. При этом предполагается, что существует доступный сервер RARP или DHCP/BOOTP. Если используется статическая DHCP-запись, BOOTP- или RARP-сервер, то необходимо знать MAC-адрес, используемый коммутатором для запроса. Для определения последнего адреса модуля 1 (т.е. блока Supervisor) используется команда **show module**

Для IOS-коммутаторов второго уровня адрес может быть получен через протокол DHCP/BOOTP в том случае, если устройство настроено на автоматическое конфигурирование (autoconfig). Автоматическое конфигурирование активизируется с помощью команды **service config**. Если автоматическое конфигурирование включено, то коммутатор игнорирует любые параметры IP-конфигурации, заданные вручную.

Системы COS
```
show module
set interface sc0 0.0.0.0
reset
```

Система IOS	`service config`
	(режим глобальной конфигурации)
	`reload`
	(в режиме привилегированного пользователя)

Для COS-коммутаторов можно обновить или освободить DHCP-адрес с помощью команды `set interface sc0 dhcp release` или `set interface sc0 dhcp renew`.

Внимание!

Настройка служб позволяет загружать полную конфигурацию для коммутатора автоматически. Эта операция в среде маршрутизаторов называется *autoinstall* (автоустановка). Для полной настройки функции автоматического конфигурирования также требуется, чтобы конфигурационный файл был доступен на TFTP-сервере. Более подробную информацию по автоматическому конфигурированию содержит страница web-сайта корпорации Cisco `www.cisco.com/univercd/cc/td/doc/product/lan/c2900xl/29_35mc/scg/swgsysb.htm#xtocid100003`.

Конфигурирование стандартного шлюза

Поскольку может понадобиться доступ к коммутатору из различных сетей в задачной среде, важно настроить адрес шлюза или стандартный маршрут для получения доступа к устройству третьего уровня, а через него — к другим сетям.

1. Конфигурирование стандартного шлюза (*необязательный, но рекомендуемый этап*)

Система COS	`set ip route default` *gatewayaddress*
Система IOS	`ip default-gateway` *gatewayaddress* (режим глобальной конфигурации)

Адресом шлюза является IP-адрес интерфейса третьего уровня, который служит в качестве маршрутизатора для трафика, генерируемого коммутатором. Для того чтобы просмотреть адреса стандартных шлюзов, для COS-устройств используется команда `show ip route`, а для IOS-устройств — команда `show ip route default`.

Внимание!

Параметр `default` в COS-конфигурации представляет собой IP-псевдоним (alias) и поэтому не может быть сохранен. Вместо ключевого слова `default` можно использовать адрес 0.0.0.0.

Использование DNS-служб или таблиц адресов станций

Каждый коммутатор Catalyst в случае, если настроена соответствующая служба, способен преобразовывать общие имена, такие, как URL или полностью указанные доменные имена, в IP-адреса. Такую службу предоставляет DNS-сервер (*Domain Name System — система доменных имен*) или локальная таблица адресов станций (host table). Стандартно на

IOS-коммутаторах DNS-службы включены, но сервер не указан. На COS-коммутаторах необходимо включить данную службу и указать сервер. При конфигурировании коммутатора для DNS службы используются описанные ниже правила.

1. Включение DNS-службы на коммутаторе (*необязательно*).

Система COS	`set ip dns enable`
Система IOS	`ip domain-lookup` (режим глобальной конфигурации)

Эта команда настраивает коммутатор на использование DNS-службы для поиска имен. Стандартная установка службы для операционной системы IOS `ip domain-lookup` включена.

Совет

Если использование DNS-службы не предусмотрено, рекомендуется отключить DNS-поиск на IOS-коммутаторах с помощью команды `no ip domain-lookup` режима глобальной конфигурации. Эта команда предотвращает попытки коммутатора преобразовывать неверно набранные команды в какие-либо имена.

2. Указание адреса DNS-сервера (*необязательно*).

Система COS	`set ip dns server` *`serveraddress`* `[primary]`
Система IOS	`ip name-server` *`serveraddress1 [serveraddress2 ... serveraddress6]`* (режим глобальной конфигурации)

Эта команда используется для указания адресов одного или нескольких DNS-серверов. Для COS-коммутаторов, используя отдельные записи командной строки, можно внести до трех различных адресов. С помощью ключевого слова `primary` в операционной системе COS определяется первый опрашиваемый сервер. Если ответ от него отсутствует, проверяются два других адреса. Для IOS-коммутаторов можно указать до шести адресов в одной строке, при этом первый адрес является основным.

3. Определение записей для преобразования имен вручную (*необязательно*).

Система COS	`set ip alias` *`name address`*
Система IOS	`ip host` *`name address`* (режим глобальной конфигурации)

При указании имени и адреса устройства на коммутаторе имя преобразовывается в локальной таблице. При использовании локальных имен узлам DNS-служба может быть включена или выключена.

Конфигурирование HTTP-служб

Для IOS коммутаторов можно включить HTTP-сервер, который позволит администрировать коммутатор с помощью web-браузера. *Графический web-интерфейс пользователя (Graphical User Interface – GUI)* является простым вариантом администрирова-

ния, предоставляющим пользователю альтернативные возможности конфигурирования. Службы HTTP-сервера поддерживаются не на всех платформах.

1. Конфигурирование HTTP-службы для коммутатора (*необязательно*)

Система COS	нет
Система IOS	[no] ip http server (режим глобальной конфигурации)

Команда IOS `ip http server` стандартно присутствует в конфигурации. Отключить сервер можно с помощью команды `no`.

Внимание!

Начиная с операционной системы Cisco IOS версии 11.3, HTTP-сервер стандартно включен. Некоторые коммутаторы содержат ошибку CSCdi93862, которая позволяет пользователям получить доступ к HTTP-службам на привилегированном уровне, минуя ввод пароля. Первоначально эта проблема решалась путем отключения HTTP-служб. Более подробные сведения приведены на странице www.cisco.com/warp/public/707/IOS_httplevel_pub.html.

Пример конфигурирования функции

Ниже приводится пример типичной конфигурации IP-адреса, шлюза и DNS-серверов для коммутатора в административной сети VLAN 986. Для IOS-коммутатора в этом примере служба HTTP-сервера отключена.

Пример конфигурации для Catalyst OS

```
Console (enable)> set interface sc0 986 10.1.1.5/24
Console (enable)> set ip route default 10.1.1.1
Console (enable)> set ip dns enable
Console (enable)> set ip dns server 10.1.1.254
```

Пример конфигурации для операционной системы IOS блока Supervisor.

```
Switch(config)#interface vlan 986
Switch(config-subif)#ip address 10.1.1.5 255.255.255.0
Switch(config-subif)#management
Switch(config-subif)#ip default-gateway 10.1.1.1
Switch(config)#ip name-server 10.1.1.254
Switch(config)#no ip http server

Switch(config)#end
Switch(config)#copy running-config startup-config
```

3.3: пароли и их восстановление

- Пароли обеспечивают некоторый уровень защиты коммутатора, предотвращающий несанкционированное использование.
- Коммутаторы Catalyst стандартно имеют два уровня паролькой защиты: пользовательский и привилегированный.

- Для обеспечения более строгой защиты пароли привилегированного уровня шифруются.
- В каждой операционной системе имеется процесс восстановления паролей, предоставляющий доступ к устройству в случае потери пароля.

Конфигурирование паролей

1. Конфигурирование пароля пользовательского уровня (*необязательно, но настоятельно рекомендуется*).

Система COS	`set password`
Система IOS	`login` `password` *password* (обе команды вводятся в режиме конфигурирования линии)

Пароль уровня пользователя предотвращает доступ неавторизованных лиц к *интерфейсу командной строки* (*CLI*) из Telnet- или консольного сеанса. В COS-устройствах команда `set password` активизирует приложение, запрашивающее у пользователя текущий и новый пароли. В устройствах под управлением операционной системы IOS на каждой линии (`con0` или `vty 0 4`) должны быть заданы команды `login` и `password`. Команда `login` является стандартной для Telnet-линий IOS-коммутаторов и препятствует входу пользователя через службу Telnet до тех пор, пока пароль не будет установлен.

Внимание!

На IOS-коммутаторе существует возможность задать различные пароли пользовательского уровня для разных линий, таких, как Telnet- и консольное соединение. В COS-коммутаторах пользовательский пароль всегда одинаков независимо от того, каким образом осуществляется доступ (через службу Telnet или консоль).

2. Конфигурирование пароля привилегированного уровня (*необязательно, но настоятельно рекомендуется*).

Система COS	`set enablepass`
Система IOS	`enable secret` *password* (режим глобальной конфигурации)

Пароль привилегированного уровня предотвращает доступ неавторизованных лиц к соответствующему уровню, на котором могут вноситься изменения в конфигурацию коммутатора и осуществляться другие функции. В COS-устройствах команда `set enablepass` активизирует приложение, запрашивающее у пользователя текущий и новый пароли. В IOS-устройствах для установки пароля используется команда `enable secret`, за которой следует пароль.

Внимание!

В COS-устройствах шифруются пароли обоих уровней. В IOS-устройствах стандартно шифруется только секретный, привилегированный пароль.

Пример конфигурирования функции

В примере приводится типичная конфигурация для установки пользовательского и привилегированного паролей в COS- и IOS-коммутаторах.

Пример конфигурации для операционной системы Catalyst OS.

```
Console (enable)>
Console (enable)>set enablepass
Enter old password: oldenablepass
Enter new password. san-fran
Retype new password: san-fran
Password changed
Console (enable)> set password
Enter old password. oldpass
Enter new password: cisco
Retype new password: cisco
Password changed
Console (enable)>
```

Пример конфигурации для операционной системы Supervisor IOS

```
Switch(config)#enable secret san-fran
Switch(config)#line vty 0 4
Switch(config-line)#password cisco
Switch(config-line)#line con 0
Switch(config-line)#login
Switch(config-line)#password cisco
Switch(config-line)#end
Switch#copy running-config startup-config
```

Восстановление пароля на COS-устройстве

Если пользователь забыл или утерял пароль, существует возможность получить доступ к коммутатору, используя процесс восстановления пароля для COS-устройства. COS-процедура является простейшей из всех процедур восстановления паролей.

В течение первых 30 секунд, после того как CLI-интерфейс коммутатора стал доступным, все пароли обнулены и пользователям необходимо только нажать клавишу <Enter> в ответ на запрос пароля (в том числе и когда отображается приглашение к процессу установки пароля). Ниже приводится описание этапов этой процедуры.

1. Отключить и включить устройство, предварительно подключив к консольному порту коммутатора консольное устройство.
2. Как только коммутатор станет доступным через консоль, нажать клавишу <Enter> для получения доступа к приглашению пользователя.

Внимание!

Должно появиться сообщение о том, что устройство доступно, однако оно может быстро выйти за границы видимой области экрана. Поскольку важно своевременно завершить следующий этап, можно нажимать клавишу «Enter» через каждые несколько секунд, пока не появится приглашение пользователя.

3. После получения доступа к приглашению пользователя с помощью команды **enable** и нажатия клавиши <Enter> в ответ на запрос пароля осуществляется вход в привилегированный режим.
4. В привилегированном режиме необходимо изменить пользовательский пароль с помощью команды **set password** и нажимать клавишу <Enter> в ответ на каждый запрос.
5. В привилегированном режиме необходимо изменить пользовательский пароль с помощью команды **set enablepass** и нажимать клавишу <Enter> в ответ на каждый запрос.

К этому моменту все пароли установлены в стандартное значение (т.е. отсутствуют). Поскольку время критично для этапов 2–5, наилучшим способом будет обнуление паролей, а затем после этапа 5 — повторная установка паролей в желаемые значения с использованием команд **set password** и **set enablepass**.

Совет

Поскольку в распоряжении пользователя имеется ограниченное время на завершение этого процесса, рекомендуется создать текстовый файл с командами и символами возврата каретки, необходимыми для завершения процесса. Затем после получения приглашения на вход, чтобы завершить процесс восстановления пароля, можно выполнить операцию вставки через меню терминального приложения для такого текстового файла.

Пример конфигурирования функции

В примере демонстрируется типичный процесс восстановления паролей для COS-коммутатора.

```
System Bootstrap, Version 5.3(1)
Copyright (c)1994-1999 by Cisco Systems, Inc.
c6k_sup1 processor with 65536 Kbytes of main memory
Autoboot executing command: "boot bootflash:cat6000-sup9k.6-1-1c.bin"
 Uncompressing file: ##################################################
##################################################################
##################################################################
System Power On Diagnostics
DRAM Size. ...  ..... . 64 MB
Testing DRAM.....Passed
Verifying Text segment..... Passed
NVRAM Size .....512 KB
Saving NVRAM .  .....  .Done
Testing NVRAM  .. .....  ..Passed
Restoring NVRAM .....Done
Level2 Cache .....Present
Testing Level2 Cache .....  .Passed

System Power On Diagnostics Complete

Boot image: bootflash:cat6000-sup9k.6.1.1(c).bin
Running System Diagnostics from this Supervisor (Module 1)
This may take up to 2 minutes ...please wait

Cisco Systems Console
Enter password:  (нажмите <Enter>)
```

```
2000 Jan 09 23:09:27 %SYS-1-SYS_NORMPWRMGMT:System in normal power
management operation
2000 Jan 09 23:09:27 %SYS_S-MOD_PWRON:Module 3 powered up
2000 Jan 09 23:09:34 %SYS_S-MOD_OK:Module 1 is online

Console> enable
Enter password : (нажмите <Enter>)
Console>(enable) set password
Enter old password: (нажмите <Enter>)
Enter new password: (нажмите <Enter>)
Retype new password: (нажмите <Enter>)
Password changed.

Console> (enable) set enablepass
Enter old password: (нажмите <Enter>)
Enter new password: (нажмите <Enter>)
Retype new password: (нажмите <Enter>)
Password changed.

Console> (enable)

Console> (enable) set password
Enter old password: (нажмите <Enter>)
Enter new password: (введите новый пароль)
Retype new password: (введите новый пароль)
Password changed

Console> (enable) set enablepass
Enter old password: (нажмите <Enter>)
Enter new password: (введите новый пароль)
Retype new password: (введите новый пароль)
Password changed
```

Восстановление пароля на IOS-устройствах: процедура №1

Процедура восстановления пароля № 1 предназначена для коммутаторов серий 2900/3500XL, 2950 и 3550. Процесс восстановления может использоваться для получения доступа к устройству тогда, когда пользователь забыл или утерял пароль, а также в том случае, если требуется пропустить загрузку конфигурационного файла.

Для того чтобы восстановить потерянный IOS пароль, необходимо остановить процесс загрузки, а затем загрузить IOS-коммутатор без использования конфигурационного файла. В процессе загрузки коммутатора без конфигурационного файла пароли отсутствуют, и пользователь может войти в привилегированный режим. В привилегированном режиме можно скопировать конфигурационный файл в активную память, а затем изменить и сохранить пароли. Для выполнения процесса восстановления необходимо осуществить описанные ниже этапы.

1. Подключить устройство к консоли коммутатора. Проверить связь, а затем извлечь из коммутатора шнур питания.
2. Нажать кнопку выбора режима (*mode*) и, удерживая ее, вставить шнур. Отпустить кнопку после того, как светодиодный индикатор над портом /x будет гореть, но крайней мере, 2 секунды.

3. Пользователю будет предоставлена информация, указывающая на то, что flash-инициализация прервана. После получения сведений необходимо в приглашении ввести команду **flash_init**.
4. Ввести команду **load_helper**.
5. Необходимо получить список содержимого flash устройства с помощью команды **dir flash:** (двоеточие **:**; является обязательным!)
6. Переименовать существующий файл в config.text с помощью команды **rename flash:config.text flash:config.old**.
7. Продолжить процесс загрузки с помощью команды **boot**.
8. Ответить **n** на вопрос о входе в режим настройки (setup mode).
9. Нажать клавишу <Enter> для доступа к пользовательскому режиму и войти в привилегированный режим с помощью команды **enable**.
10. Переименовать конфигурационный файл обратно в config.text с помощью команды **rename flash:config.old flash:config.text**.
11. Скопировать конфигурационный файл в активную память с помощью команды **copy flash:config.text system:running-config**.
12. Войти в конфигурационный режим с помощью команды **configure terminal**.
13. Изменить секретный пароль и пароль линии, как описывалось выше.
14. Сохранить конфигурацию.

Пример использования функции

В примере демонстрируется типичная процедура восстановления пароля №1 для IOS-коммутаторов.

Система была остановлена до инициализации файловой системы flash-устройства. С помощью приведенных ниже команд инициализируется файловая система устройства flash и завершается загрузка программного обеспечения операционной системы.

```
flash_init
load_helper
boot

flash_init
load_helper
dir flash:
  Directory of flash:
2 -rwx 843947 May 01 1993 05:02:18 c2900XL-mg-12.2.8.bin
4 drwx 2776 May 01 1993 01:23:24 html
66  rwx 120 Jan 01 1970 00:01:19 env_vars
68 -rwx 7296 May 01 1993 06:55:51 config.text
1728000 bytes total (1456704 bytes free)
rename flash:config.text flash:config.old
boot
Continue with the configuration dialog?[yes/no] N
Switch>enable
Switch#rename flash:config.old flash:config.text
Switch# copy flash:config.text system:running-config
```

```
Switch#configure terminal
Switch(config)#enable secret пароль
Switch(config)#line vty 0 4
Switch(config-line)#password пароль
Switch(config)#line con 0
Switch(config-line)#password пароль
Switch(config-line)#end
Switch#copy running-config startup-config
```

Восстановление пароля на IOS-устройствах: процедура №2

Процедура восстановления пароля №2 предназначена для коммутаторов серии 6000, использующих операционную систему IOS. Процесс восстановления может использоваться в ситуации, когда пользователь забыл или утерял пароль, а также в случае, если требуется пропустить загрузку конфигурационного файла.

Для того чтобы восстановить IOS-пароль, необходимо остановить процесс загрузки процессора маршрутизации, а затем загрузить IOS-коммутатор без использования конфигурационного файла. В процессе загрузки коммутатора без конфигурационного файла пароли отсутствуют, и пользователь может войти в привилегированный режим. В привилегированном режиме можно скопировать конфигурационный файл в активную память, а затем изменить и сохранить пароли. Для выполнения процесса восстановления необходимо осуществить описанные ниже этапы.

1. Подключить устройство к консоли коммутатора, отключить и включить питание устройства.
2. Просмотреть информацию, отображаемую на консоли. Когда появится сообщение "%OIR-6-CONSOLE. Changing console ownership to route processor" (изменение владельца консоли на процессор маршрутизации), необходимо ввести спецпоследовательность (break sequence) из эмулятора терминала (как правило, используется комбинация клавиш <Ctrl>+<Break>).
3. Должно появиться приглашение `rommon1>`, в котором необходимо ввести команду `confreg 0x2142`, сообщающую коммутатору о необходимости игнорировать текущую конфигурацию.
4. Ввести команду `reset` в приглашении `rommon2>` для перезагрузки коммутатора и перезапуска процесса загрузки.
5. Ответить `n` на вопрос о входе в режим настройки.
6. Нажать клавишу <Enter> для доступа к приглашению `Router>` и ввести команду `enable` для входа в привилегированный режим.
7. В приглашении `Router#` скопировать загруженную конфигурацию в действующую с помощью команды `copy startup-config running-config`.
8. Войти в режим глобальной конфигурации с помощью команды `configure terminal`.
9. Изменить линейный и секретный пароли, как указывалось ранее.
10. Перустановить конфигурационный регистр с помощью команды `config-register 0x2102`.

11. Выйти из режима настройки с помощью команды end.
12. Сохранить конфигурацию с помощью команды copy running-config startup-config.

Пример использования функции

В примере демонстрируется типичная процедура восстановления пароля №2 для IOS-коммутаторов.

```
%SYS-5-CONSOLE_OWNERSHIP_CHANGE: Changing console ownership to route processor
ieeue break
rommon1>confreg 0x2142
rommon2>reset
<информация, отображаемая коммутатором запуска>
Continue with the configuration dialog?[yes/no]: N
Router>enable
Router# copy startup-config running-config
Router#configure terminal
Router(config)#enable secret newpassword
Router(config)#line vty 0 4
Router(config-line)#password newpassword
Router(config)#line con 0
Router(config-line)#password newpassword
Router#config-register 0x2102
Router(config-line)#end
Router#copy running-config startup-config
```

3.4: управление модулями

- Во многих устройствах имеются несколько отсеков (blade) или модулей, которые используются для служб коммутации.
- Некоторые из таких модулей имеют собственные операционные системы и могут быть доступны непосредственно для конфигурирования.
- Большинство модулей можно отключить, включить или перезагрузить индивидуально.
- В некоторых коммутаторах существует возможность отключить питание какого-либо модуля.
- В COS-коммутаторах пользователь может просмотреть конфигурацию каждого модуля в отдельности.
- Для COS-коммутаторов существует возможность очистить конфигурацию какого-либо одного модуля.

Просмотр информации о модулях

В обеих системах — как в IOS, так и в COS — существует возможность использовать модульные структуры для создания более гибкой конфигурации коммутаторов. Для того чтобы просмотреть информацию о модулях, установленных в коммутаторе, используется одна из следующих команд.

Система COS	`show modules`
Система Supervisor IOS	`show module all` (привилегированный режим)
Система IOS второго уровня	`show hardware` или `show version` (привилегированный режим)

С помощью указанных выше команд отображаются сведения по аппаратному обеспечению или модулям коммутатора.

Доступ к модулям

Конфигурирование модулей и портов настраивается через главный CLI-интерфейс коммутатора. Однако к некоторым модулям, таким, как *функциональная плата многоуровневой коммутации (Multilayer Switch Feature Card — MSFC)* и ATM-интерфейс коммутатора 2900MXL, требуется индивидуальный доступ и отдельная настройка конфигурации. Чтобы получить доступ к таким интерфейсам, используется команда `session`.

Система COS	`session` *mod#*
Система IOS	`session slot#` (привилегированный режим)

Параметры *slot#* или *mod#* указывают, в каком гнезде или модуле установлена плата. Для коммутатора 2900MXL указывается гнездо 1 или 2. Для модулей третьего уровня, RSM или MSM, в коммутаторе Catalyst 4000, 5000 или 6000 значение данного параметра соответствует номеру модуля, в которым установлена плата. Для *функциональной платы коммутации маршрутов (Route Switch Feature Card — RSFC)* и MSFC-платы это число равно 15 в гнезде 1 и 16 — в гнезде 2.

Внимание!

Ко многим из перечисленных выше модулей можно получить доступ с помощью консольного порта непосредственно в соответствующем отсеке. Для консольного доступа к MSFC-плате коммутатора Catalyst 6000 используется команда `switch console`.

Перезагрузка модулей

Существует возможность перезагружать модули индивидуально. Следовательно, можно перезагрузить (jumpstart) группу портов, не перезагружая весь коммутатор.

Система COS	`reset` *mod#*
Система IOS	`power cycle module slot` (привилегированный режим)

Команда `reset` приводит к отключению и последующему восстановлению питания всего модуля, а также вынуждает модуль пройти процедуру *POST-проверки (Power-On Self-Test — самотестирование при включении питания)* при загрузке. В некоторых IOS-коммутаторах такая возможность недоступна. В таких коммутаторах можно перезагрузить порт с помощью команд `shutdown` и `no shutdown`.

Включение и отключение питания модулей

Для модульных IOS- и COS-коммутаторов можно отключить питание какого-либо модуля. Отключение питания приводит к выключению модуля и всех его портов. Если коммутатор перезагружается или проходит цикл отключения питания, модуль остается выключенным. Такое состояние может оказаться полезным для поиска и устранения неисправностей при загрузке или в случае, если источник питания не способен поддерживать всю нагрузку коммутатора.

Система COS	`set module power down` *mod*
Система IOS	`no power enable module` *slot* (режим глобальной конфигурации)

С помощью указанных команд отключаются модули коммутатора. В IOS-коммутаторах конфигурация модуля не сохраняется и, если коммутатор перезагружается, все конфигурационные записи для отключенного модуля теряются. Для включения модулей используются команды, приведенные ниже.

Система COS	`set module power up` *mod*
Система IOS	`power enable module` *slot* (режим глобальной конфигурации)

Отображение конфигурации модулей

Просмотреть конфигурацию определенного модуля в COS-коммутаторах можно, используя приведенные ниже команды.

Система COS	`show configuration` *mod#* `(all)`
Система IOS	нет

Для указания определенного модуля используется его номер. Если параметр `all` не указан, то отображаются только нестандартные параметры.

Удаление конфигурации модулей

Очистить конфигурацию определенного модуля COS-коммутатора можно с помощью приведенной ниже команды.

Система COS	`clear config` *mod#* `(all)`
Система IOS	нет

Для указания определенного модуля при удалении конфигурации используется параметр *mod#*. Эта команда особенно полезна в случае, если плата удаляется и заменяется другой. Например, если удаляется восьмипортовый Gigabit Ethernet-модуль и заменяется шестнадцатипортовым модулем, необходимо удалить конфигурацию перед тем, как активизируется новый модуль.

3.5: управление файлами и параметры загрузки

- В операционных системах Cisco имеется множество файлов, требующих администрирования.
- Управление файлами (file management) включает в себя управление конфигурационными файлами и файлами операционной системы.
- Для COS-коммутаторов не требуется сохранять конфигурационные изменения.
- Команды файловой системы заменяют многие более ранние команды управления файлами.
- Команды файловой системы позволяют просматривать и классифицировать все файлы, включая файлы на удаленных серверах.
- Команды файловой системы позволяют копировать файлы с помощью информации о путях к ним, делая ненужными системные подсказки.
- Платформы производства Cisco поддерживают различные типы файловых систем на основе flash-памяти.
- При монтировании различных файлов во flash-память важно сконфигурировать коммутатор для загрузки соответствующего файла, содержащего параметры загрузки.

Внимание!

Как COS-, так и IOS-коммутаторы имеют новый набор команд файловой системы, которые облегчают управление файлами. Корпорация Cisco обозначает файловую систему как *IFS*, или *IOS file system (файловая система IOS)*. Эта файловая система представляет собой чрезвычайно мощный способ управления файлами внутри коммутаторов и в удаленных системах. Поскольку данная система появилась недавно, многие специалисты не знакомы с ее командами. Для обеспечения обратной совместимости используется множество псевдонимов, преобразовывающих новые команды управления файлами в более ранние. Список старых команд и их IFS-эквивалентов приводен в табл. 3.3 в конце текущего раздела

Команды перемещения объектов в файловых системах

1. Просмотреть доступные устройства файловой системы можно с помощью следующих команд:

Система COS	`show flash devices`
Система IOS	`show filesystems` (привилегированный режим)

Эта команда представляет список файловых систем, доступных на данном устройстве. COS-команда перечисляет записываемые устройства. IOS-команда отображает общий размер и объем свободного пространства в файловой системе в байтах, тип файловой системы, флаги и псевдонимы, которые используются для доступа к файловой системе. Для IOS-устройств поддерживаемые типы файловых

систем включают в себя flash, энергонезависимую память (*nonvolatile random-access memory — NVRAM*) и сетевой доступ к хранилищу (а также некоторые другие, например, файловые системы постоянной памяти ROM, содержащие микрокод). В табл. 3.1 перечислены некоторые доступные файловые системы. Следует заметить, что не все файловые системы доступны на всех платформах.

Таблица 3.1. Файловые системы устройств Cisco

Префикс	Файловая система
system:	Содержит системную память, включая действующую конфигурацию
nvram:	Энергонезависимая память, содержащая загрузочную конфигурацию
flash:	Flash-память, в которой, как правило, размещена операционная система IOS. Эта система является стандартной или начальной файловой системой для перемещения по файловым системам. Префикс *flash:* доступен на всех платформах. Для платформ, в которых устройство с именем *flash:* отсутствует, этот префикс является псевдонимом к устройству с именем *slot0:*. Следовательно, префикс *flash:* можно использовать для обращения к главной области хранения данных flash-памяти на всех платформах
bootflash:	Загрузочная flash-память. Обычно в ней располагается образ операционной системы RxBoot IOS
supbootflash:	Загрузочная flash-память для процессора управления сетью (*Network Management Processor — NMP*) блока Supervisor. Начальная точка загрузки операционной системы Catalyst
slot0:	Первая PCMCIA-карта flash-памяти
tftp:	Сетевой сервер TFTP (Trivial File Transfer Protocol — простой протокол передачи файлов)
ftp:	Сетевой сервер FTP (File Transfer Protocol — протокол передачи файлов)
slave-nvram:	NVRAM-память на резервном модуле Supervisor, использующем собственную систему IOS
slave-supbootflash:	Загрузочная flash-память для NMP процессора резервного модуля Supervisor
slave-bootflash:	Внутренняя flash-память в резервной MSFC-плате, использующей собственную систему IOS
slave-slot0:	Первая PCMCIA-карта в резервном модуле Supervisor
null:	Несуществующий получатель копий (null destination for copies). Копировать удаленный файл на несуществующее устройство можно с целью определения размера файла или проверки его целостности
rcp:	Сетевой сервер RCP (*Remote Copy Protocol — протокол удаленного копирования*)

2. Изменение стандартного каталога файловой системы.

Система COS	`cd [filesystem:]`
Система IOS	`cd [filesystem:]` (привилегированный режим)

Указанные команды используются для перехода к определенной файловой системе или каталогу, находящемуся внутри файловой системы. Перемещаясь к оп-

ределенному местоположению файлов, можно использовать команды файловой системы без указания параметра *filesystem:*. Например, если используется команда `dir` без параметра *filesystem:*, то система использует стандартный каталог или каталог, указанный командой `cd`. Стандартным каталогом файловой системы является каталог с именем *Flash*.

3. Имя текущего каталога.

Система COS	`pwd`
Система IOS	`pwd` (привилегированный режим)

С помощью этой команды выводится на печать или отображается на экране имя рабочего каталога. Команда позволяет определить стандартный каталог файловой системы, ее также можно использовать для того, чтобы проверить, был ли совершен переход в соответствующий каталог с помощью команды `cd`.

4. Отображение сведений о файлах.

Система COS	`dir` *[/m/]device:][filename]* `[all | deleted | long]`
Система IOS	`dir [/all]` *[filesystem:][path/filename]* (привилегированный режим)

С помощью этой команды отображается каталог структуры стандартного каталога, указанный командой `cd`. Параметры `/all` для IOS- или `all` для COS-коммутаторов позволяют отобразить все файлы, включая те, которые не были окончательно удалены из файловой системы. Для COS параметр `deleted` позволяет отобразить только удаленные файлы, которые не были окончательно удалены. Параметр `long` позволяет отображать имена в длинном формате. Для того чтобы задать номер модуля, в COS-устройстве можно указать параметр в перед именем устройства. Кроме того, используя параметр *filesystem:* или *device:*, можно задать файловую систему. Если необходимо просмотреть один файл, следует указать также путь и имя этого файла. Для того чтобы отобразить группу файлов с одинаковыми первыми символами в именах, в качестве символа подстановки можно использовать звездочку (*). Чтобы получить список файлов какой-либо доступной локальной файловой системы, можно использовать команды, приведенные ниже.

Система COS	`show filesystem:`
Система IOS	`show filesystem:` (привилегированный режим)

С помощью этой команды отображается содержимое файловой системы. Указанная команда подобна команде `dir`, однако отображаемая информация форматируется иначе. Команда не позволяет вывести на экран отдельные файлы или удаленные файловые системы.

5. Просмотр информации о локальном или недокальном файле

Команда позволяет просмотреть сведения о каком-либо файле в отдаленной или локальной файловой системе. В отображаемую информацию включаются тип образа и размер.

Система COS	нет
Система IOS	`show file information` *filesystem:path* (привилегированный режим)

6. Просмотр содержимого локального или удаленного файла.

Система COS	`show file` *[device:] filename* [dump]
Система IOS	`more` [`/ascii` | `/binary` | `/ebcdic`] *filesystem:path* (привилегированный режим)

Команда используется для просмотра содержимого локального файла или файла на удаленной системе. Параметры **ascii**, **binary** и **ebcdic** позволяют указать тип формата, в котором необходимо представить этот файл. Для системы COS стандартным является ASCII-формат. Если используется параметр **dump**, то файл отображается в двоичном формате. Параметры *filesystem:path* позволяют указать определенный файл на доступной файловой системе. Например, с помощью команды **more /ascii flash:myconfig.txt** отображается файл **myconfig.txt** в формате ASCII, расположенный в текущем flash-устройстве.

Удаление файлов из flash-памяти

Устройства коммутации Cisco имеют три различных класса файловых систем. В каждой системе имеется собственный способ решения проблемы удаления и окончательного стирания файлов из файловой системы flash. В табл. 3.2 приведены три типа файловых систем и платформы, на которых они используются.

Таблица 3.2. Типы файловых систем коммутаторов

Тип файловой системы	Платформы
Класс A	Catalyst 5000, 5500, 4000, 2948G, 4908G
Класс B	Catalyst 2948G-L3, 4908G-L3
Класс C	Catalyst 2900/2900XL, 8550, 2950

1. Удаление файла из flash-памяти.

Система COS	`delete` *[filesystem:] filename*
Система IOS	`delete` *[filesystem:] filename* (привилегированный режим)

С помощью указанной команды удаляется какой-либо файл из flash-памяти в файловой системе любого класса. Для файловых систем классов A и B файл маркируется как удаленный и показывается только при использовании команды **dir /all**. С помощью команды **undelete** можно восстановить файл, помеченный как удаленный. В файловых системах класса C команда **delete** окончательно удаляет файл из системы. Типом используемой файловой системы должен быть Flash.

2. Восстановление удаленного файла.

Система COS	`undelete index [filesystem:]`
Система IOS	`undelete index [filesystem:]` (привилегированный режим)

В системе класса A удаленный файл может быть восстановлен с помощью команды `undelete`. Необходимо указать индексный номер файла (`index`), который отображается с помощью команды `dir /all`. Если файл расположен не в данном рабочем каталоге, определенном с помощью команды `pwd`, то можно задать дополнительный параметр `[filesystem:]`.

3. Окончательное удаление файла из flash-памяти класса A.

Система COS	`squeeze filesystem:`
Система IOS	`squeeze filesystem` (привилегированный режим)

Если требуется окончательно удалить файл из системы класса A, необходимо сжать файловую систему с помощью команды `squeeze` (сжать). Эта команда окончательно удаляет из файловой системы любой файл, который был помечен как удаленный.

4. Удаление файла из flash- или NVRAM-памяти.

Система COS	нет
Система IOS	`delete [flash:/filename	bootflash:/filename	nvram:/filename]` (привилегированный режим)

Чтобы удалить файл из flash-устройства класса B, нужно использовать команду `delete`. При удалении файла из flash-устройства класса B он остается во flash-памяти и занимает в ней некоторое пространство. Для окончательного удаления файла из файловой системы класса B ее необходимо переформатировать с помощью команды, приведенной ниже. Поскольку при этом удаляются все файлы, следует сохранить файлы операционной системы, а затем после форматирования устройства скопировать их обратно в память.

Система COS	`format filesystem:`
Система IOS	`format filesystem:` (привилегированный режим)

В устройствах классов A и C также можно удалить все файлы и переформатировать устройство с помощью команды `format`.

Перемещение системных файлов

Как и в большинстве компьютерных систем, важной функцией является перемещение файлов из одного каталога другой. Для перемещения системных файлов можно использовать команду `copy`. Эта команда, используемая с указанием параметров пути, перемещает системные файлы. Результаты некоторых перемещений файлов уникальны — например, когда файл копируется в файл текущей системной

конфигурация system:running-configuration, происходит слияние файлов. В этом разделе обсуждаются некоторые общие команды группы copy и результаты их выполнения. В целом существует возможность перемещать файлы в файловые системы, которые позволяют осуществлять запись в систему. Структура команды copy: copy [/erase] *source location destination location*. Параметры начального (*source-location*) и конечного расположения (*destination-location*) могут соответствовать любой записываемой файловой системе и пути. Используя параметр /erase, всегда можно стереть файл в файловой системе получателя, допускающей запись, до того как файл-источник будет скопирован. Источником может быть любая файловая система, содержащая файлы, которые необходимо переместить. Со всеми этими командами можно указывать адрес и имя файла или оставить поля пустыми, а система запросит необходимые сведения.

1. Сохранение файла активной конфигурации для использования при загрузке.

Система COS	Не требуется (осуществляется автоматически)
Система IOS	`copy system:running-config nvram:startup-config` (привилегированный режим)

С помощью этой команды активная системная конфигурация копируется в файл загрузочной конфигурации. Когда какие-либо данные копируются в файл начальной конфигурации (nvram:startup-configuration), он полностью перезаписывается, т. е. любая информация, которая была в файле, полностью теряется и заменяется данными файла-источника. Файл начальной конфигурации загружается при включении системы. Для COS-устройств эта операция выполняется автоматически.

2. Копирование файла в файл активной конфигурации.

Система COS	`copy [[m/]device:][filename] config`
Система IOS	`copy source system:running-config` (привилегированный режим)

С помощью этой команды файл копируется в текущую рабочую конфигурацию. Источником/устройством может быть любое устройство, содержащее текстовый файл с конфигурационными параметрами, отформатированными с соблюдением соответствующих синтаксических правил. Когда файлы копируются в действующую конфигурацию, они объединяются с текущей конфигурацией. Т. е. если какой-либо конфигурационный параметр (такой, как адрес) существует в обоих файлах — работающей и загрузочной конфигурации, в текущей конфигурации этот параметр будет заменен параметром, который копируется из файла-источника. Если конфигурационный параметр существует только в источнике, он добавляется к действующей конфигурации. В случае, когда какой-либо параметр существует в действующей конфигурации, но не модифицируется в источнике, изменений в действующей конфигурации не будет. Исходным может быть файл в любом каталоге, включая файл на TFTP-сервере, FTP-сервере или текстовый файл, записанный во flash-память.

3. Сохранение файла на TFTP-сервере.

Система COS	`copy [lm/]device:[filename] tftp`
Система IOS	`copy source tftp://address/filename` (привилегированный режим)

Эта команда позволяет сохранить на TFTP-сервере, указанном в параметре адреса получателя, любой читаемый файл из IFS-источника. Если пользователь не ввел имя файла и адрес, то система запросит эти сведения.

4. Сохранение файла во flash-память.

Система COS	`copy [lm/]device:[filename] flash-filesystem://path/filename`
Система IOS	`copy source flash-filesystem://path/filename` (привилегированный режим)

Существует возможность скопировать файл в любую файловую систему Flash маршрутизатора с помощью команды *copy*. Некоторые записываемые файловые системы, такие, как система класса A, допускают создание каталогов файлов, а также запись в них. Команда `copy` позволяет перемещать файлы в файловую систему Flash. Файлы, перемещаемые во flash-память, как правило, являются файлами IOS, однако можно использовать flash-память для хранения любого файла, который может поместиться в свободном пространстве. Действительно, после того как файл был помещен во flash-память, устройство может быть сконфигурировано в качестве TFTP-сервера и затем предоставлять данный файл другим устройствам. Более подробная информация по конфигурированию маршрутизатора или коммутатора в качестве TFTP-сервера приведена в командной части раздела "1.2: программное обеспечение межсетевой операционной системы Cisco (IOS)".

Загрузочные параметры файловой системы

1. Указание образа операционной системы для загрузки из файловой системы Flash.

Системы COS	`set boot system flash device:[filename] [prepend] [mod]`
Системы IOS	`boot system flash flash-filesystem:/directory/filename` (режим глобальной конфигурации)

По умолчанию коммутаторы загружают первый доступный образ в стандартном flash-устройстве. Если во flash-памяти присутствуют несколько файлов и требуется загрузить не первый файл, то необходимо указать, какой файл будет использоваться в качестве IOS-образа. Для COS-устройств можно манипулировать порядком использования файлов с помощью параметра `prepend` указанной выше команды. Файл, предназначенный для использования, определяется с помощью команды `boot system flash`.

2. Изменение параметра среды в конфигурационном файле для систем класса A.

Системы COS	`set boot auto-config device:directory/filename`
Системы IOS	`boot config device:directory/filename` (режим глобальной конфигурации)

Для какой-либо системы класса А можно скопировать конфигурационные файлы в файловую систему Flash. Можно также указать, что некоторые коммутаторы должны загружать конфигурацию из flash-памяти вместо файла загрузочной конфигурации, расположенного в NVRAM. Для того чтобы это сделать, необходимо прежде всего скопировать активную конфигурацию в файловую систему Flash, а затем в режиме глобальной конфигурации использовать параметр boot config с последующим именем файловой системы, местоположением и именем файла. После сохранения данной конфигурации устройство предпримет попытку загрузить эту конфигурацию из указанной точки.

Команды-псевдонимы

Поскольку набор функций новых файловых систем представляет собой третье поколение систем управления файлами для Cisco IOS и COS, команды-псевдонимы предоставлены с целью обеспечения обратной совместимости для команд, существовавших в предыдущих операционных системах. Такая совместимость позволяет пользователю применять команды управления файлами, которые были изучены в предыдущих версиях, без необходимости изучения новой структуры команд. В табл. 3.3 представлены команды-псевдонимы, IFS-эквиваленты и команды COS.

Таблица 3.3. Команды-псевдонимы для управления файлами

Команда Cisco IOS версии 10.2 и более ранних версий	Команда Cisco IOS версий 10.3—11.3	Команда Cisco IOS версии 12.0 и более поздних (IFS-) версий	COS-команды
`write terminal`	`show running-config`	`show system:running-config` или `more system:running-config`	`show config` или `write terminal`
`show config`	`show startup-config`	`show system:startup-config` или `more system:startup-config`	`show config` или `write terminal`
`write memory`	`copy running-config startup-config`	`copy system:running-config nvram:startup-config`	Нет эквивалента. Все конфигурационные изменения сохраняются автоматически
`write erase`	`erase startup-config`	`erase nvram:`	`clear config all`
`write network`	`copy running-config tftp:`	`copy system:running-config tftp://address/filename`	`Write network` или `copy config tftp`
`config memory`	`copy startup-config running-config`	`copy nvram:startup-config system:running-config`	Нет эквивалента. Не существует разницы между начальной и действующей конфигурациями

Окончание табл. 3.3

Команда Cisco IOS версии 10.2 и более ранних версий	Команда Cisco IOS версий 10.3—11.3	Команда Cisco IOS версии 12.0 и более поздних (NFS-) версий	COS-команды
`config network`	`copy tftp running-config`	`copy tftp://address/filename system:running-config`	`config network` или `copy tftp config`
`config overwrite`	`copy tftp startup-config`	`Copy tftp://address/filename nvram:startup-config`	Нет эквивалента

Официальная позиция корпорации Cisco по старым командам заключается в том, что они могут не поддерживаться в последующих версиях. Поэтому, вероятно, команды, существовавшие до версии 12.0, могут не поддерживаться в будущих версиях операционной системы Cisco IOS.

3.6: резервные блоки Supervisor

- Когда идентичное аппаратное обеспечение Supervisor устанавливается в гнезда 1 и 2 коммутаторов Catalyst серий 5500 и 6000, один блок является активным, а второй находится в режиме ожидания (standby mode).
- При сбое резервный блок Supervisor принимает на себя все функции коммутатора.
- Конфигурационные файлы и файлы операционных систем синхронизируются между коммутаторами.
- Таблицы второго уровня синхронизируются между блоками Supervisor для быстрой передачи функций управления между модулями.
- Можно управлять такими параметрами модулей, как синхронизация второго уровня и синхронизация операционной системы.
- MSFC-планы третьего уровня функционируют независимо от системы COS и должны управляться вручную или конфигурироваться для обеспечения резервирования в операционной системе IOS модуля MSFC. Более подробная информация по модулям приведена в разделах "8.4: резервирование модулей MSFC в одном устройстве" и "8.5: MSFC-резервирование с синхронизацией конфигурации".
- Коммутаторы серии 6000 обеспечивают синхронизацию как второго, так и третьего уровня внутри операционной системы, а также синхронизацию конфигурационных параметров.

После установки идентичного аппаратного и программного обеспечения блоков Supervisor в гнезда 1 и 2 коммутаторов Catalyst серий 5500 и 6000 активизируется резервирование в системе. Для включения этой функции никакие параметры не требуются. Первый блок Supervisor, подключившийся к линии, является активным, а второй переходит в режим ожидания. На резервном блоке Supervisor горит оранжевый системный индикатор, а консольный порт неактивен. Однако интерфейсы данного модуля являются активными.

Платы Supervisor можно удалять или устанавливать при включенном питании коммутатора. Когда добавляется второй блок или заменяется резервный Supervisor, плата, устанавливаемая в коммутатор, проходит процедуру диагностики, такую же, как при включении питания, но не тестирует объединительную плату (поскольку в таком случае прервался бы поток данных) и затем переходит в режим готовности. Резервный Supervisor становится активным, если возникает сбой в основном модуле Supervisor или в связи с инструкциями, вызвавшими перезагрузкой основного блока Supervisor.

Форсирование перехода на резервный блок Supervisor

1. Перезагрузка активного блока Supervisor

Система COS	`reset` *mod#* или `switch supervisor`
Система IOS	`reload` (привилегированный режим)

Для COS-устройства можно перезагрузить активный модуль и таким образом форсировать переход резервного модуля в активный режим. В команде `reset` *mod#* параметр *mod#* в таком случае был бы номером активного модуля. Для определения того, какой из модулей Supervisor является активным, используется команда `show module`. С помощью команды `switch supervisor` автоматически запускается перезагрузка активного блока Supervisor и происходит вынужденный переход коммутатора на резервный блок.

Конфигурационные изменения вносятся в активный модуль Supervisor. Эти изменения также распространяются на NVRAM-память резервного модуля. Если имеется изменение или отличие в образах Supervisor, активный Supervisor также инициирует синхронизацию образов для обеспечения совместимости функций. К синхронизации COS-образов между коммутаторами приводят несколько факторов; их описание приведено ниже.

- Изменение параметров загрузки для коммутатора.
- Перезапись выполняемого COS-образа.
- Удаление выполняемого COS-образа.
- Различные временные метки выполняемых образов, хранящихся в устройствах `bootflash:` или `slot0:`.

Коммутаторы серии 6500 с собственной системой IOS не выполняют автоматически синхронизацию образов. Следовательно, для функционирования резервирования необходимо наличие одинакового образа как на активном, так и на резервном модулях Supervisor. Для того чтобы вручную синхронизировать образы, необходимо убедиться, что на обоих модулях Supervisor используется система IOS, а затем скопировать образ с активного модуля на "ведомый".

Синхронизация IOS-образов

1. Синхронизация образов для модулей Supervisor под управлением системы IOS вручную (*обязательно*)

Система COS	Необязательно. Осуществляется автоматически
Система IOS	`copy source_device:source_filename destination_device:target_filename` (привилегированный режим)

Устройством-получателем может быть одно из следующих устройств:

- `slaveslot0:` — PCMCIA-карта на избыточном модуле Supervisor;
- `slave-supbootflash:` — загрузочная flash-память Supervisor в резервном модуле;
- `slave-bootflash:` — загрузочная flash-память платы MSFC в резервном модуле Supervisor.

При загрузке каждый модуль Supervisor проверяет конфигурационный регистр для определения того, каким образом необходимо загрузить и где искать образ. Как правило, с помощью параметров в переменных загрузки указывается образ во flash-устройстве. В операционных системах Catalyst конфигурационные регистры не синхронизируются автоматически для того, чтобы предоставить пользователю возможность индивидуально контролировать загрузку модулей. В то же время загрузочные переменные синхронизируются. Стандартно в IOS-устройствах Cisco конфигурационные регистры синхронизируются, а загрузочные переменные — нет. В загрузочных переменных указывается расположение операционной системы, файла-загрузчика (bootloader file) и конфигурационных файлов, используемых при загрузке.

Важно, чтобы оба модуля имели конфигурационные параметры, позволяющие им автоматически загружать одинаковые образы до того, как резервирование действительно будет иметь место. Стандартно в конфигурационных регистрах обеих операционных систем используются системные команды для загрузки операционной системы. Следовательно, они сконфигурированы корректно. Когда пользователь указывает расположение COS-образа, инициируется синхронизация операционных систем. Если конфигурационный регистр изменен, то необходимо вручную настроить регистр на каждом модуле (1 и 2) COS-устройства так, чтобы указанный образ загружался автоматически. IOS-устройства синхронизируют конфигурационный регистр со стандартным значением. Однако, если параметры загрузки были изменены, то сохранение конфигурации в активном блоке Supervisor не приведет к изменению загрузочных переменных резервного модуля.

Синхронизация загрузочных параметров

1. Синхронизация конфигурационного регистра (*обязательно*).

Система COS	`set boot config-register boot {rommon | bootflash | system} [module]` (привилегированный режим)
Система IOS	Не требуется. Осуществляется автоматически

При установке конфигурационного регистра в COS-устройстве с помощью параметра `module` необходимо указать, какой блок Supervisor настраивается. В процессе загрузки Supervisor загружается с одного из устройств: `rommon` (ROM-монитор, системные образы не загружаются), `bootflash` (первый файл

образа, хранящийся во flash-памяти устройства) или `system` (системный образ, указанный переменной BOOT-среды). Рекомендуется избегать использования ключевого слова `bootflash`, поскольку первый образ, хранящийся во flash-памяти, не всегда является желаемым.

2. Синхронизация местоположения загрузочного образа *(обязательно)*.

Системы COS	Не требуется. Осуществляется автоматически

Системы IOS	`redundancy`
	`main-cpu`
	`auto-sync bootvar`
	`end`
	`copy running-config startup-config`

После того как резервный блок Supervisor начнет функционировать, можно проверить его состояние с помощью команды `show module` для COS-коммутаторов или `show module all` для IOS-устройств, чтобы убедиться, что один модуль Supervisor является активным, а второй находится в режиме ожидания.

Если в коммутатор установлены два модуля Supervisor, то коммутатор функционирует в режиме резервирования для COS-коммутаторов и в высоконадежном режиме улучшенной системы для IOS-коммутаторов. Это означает, что конфигурация коммутаторов синхронизируется и если один модуль выйдет из строя, резервный возьмет на себя его функции. Однако, если резервный модуль возьмет на себя функции COS-коммутатора, он должен инициализировать порты и запустить *протокол распределенного связующего дерева (Spanning Tree Protocol — STP)* для определения состояния портов. Такое восстановление после сбоя длится около 30–40 секунд. Для более быстрого восстановления блоков Supervisor необходимо включить режим высокой надежности COS-коммутатора. В этом режиме коммутаторы также синхронизируют информацию за счет ее передачи между модулями. Некоторые функции, включая STP, несовместимы с высокоуровневыми параметрами, поэтому восстановление будет более эффективным. В случае с IOS-коммутатором процессоры второго и третьего уровней работают с одинаковым программным обеспечением и таблицы передачи, а также конфигурация, синхронизируются автоматически.

Однако в случае с COS-устройствами, прежде чем включать режим высокой надежности, необходимо учесть несколько факторов. Следует помнить, что с таким режимом несовместимы некоторые функции:

- динамические VLAN-сети;
- *протокол GARP-регистрации VLAN-сетей (GARP VLAN Registration Protocol — GVRP)*;
- защита портов;
- фильтрация протоколов.

Если требуется применить какую-либо из указанных функций, то включение службы высокой надежности невозможно. Наоборот, если службы повышенной надежности включены, то настроить эти функции будет невозможно. Отключенный режим высокой надежности не препятствует активизации резервного блока Supervisor; этот режим препятствует только синхронизации таблиц пере-

дачи и, следовательно, приводит к тому, что резервный блок Supervisor определяет всю информацию передачи после активизации.

3. Включение режима высокой надежности (*необязательно*).

Система COS	`set system highavailability {enable | disable}` (привилегированный режим)
Система IOS	нет

В стандартной конфигурации служба обеспечения высокой надежности отключена. Для включения режима высокой надежности используется ключевое слово **enable**. Если требуется отключить функцию, используется ключевое слово **disable**.

Включение проверки версий

Иногда требуется обновить образ активного блока Supervisor, не обновляя при этом образ резервного на случай восстановления прежнего образа. Еще одну функцию повышения надежности COS-устройства иногда называют *контролем версий (versioning)*. Контроль версий позволяет иметь отдельные выполняемые образы для блоков Supervisor в допустимых пределах. Например, версии 6.1(3) и 6.1(4) являются полностью совместимыми образами. Более подробная информация о контроле версий приводится в примечаниях к используемому образу операционной системы (Release Notes). Активный модуль Supervisor обменивается сведениями о версиях образов с резервным модулем и определяет, совместимы ли эти образы для включения функции высокой надежности. Чтобы включить контроль версий высокой надежности, необходимо реализовать приведенную ниже конфигурацию.

1. Включение службы повышения надежности (*обязательно*).

Система COS	`set system highavailability enable` (привилегированный режим)
Система IOS	нет

Эта конфигурация является стандартной, и ввод указанных команд требуется, только если служба повышения надежности ранее была отключена.

2. Включение контроля версий службы высокой надежности (*обязательно*).

Система COS	`set system highavailability versioning enable` (привилегированный режим)
Система IOS	нет

3.7: протокол обнаружения устройств Cisco

- Протокол CDP (Cisco Discovery Protocol — протокол обнаружения устройств Cisco) используется для идентификации непосредственно соединенных устройств корпорации Cisco.

- Протокол CDP поддерживается во всех устройствах производства Cisco.
- С помощью протокола CDP определяется адрес соседнего устройства, операционная система, VLAN-сеть, VTP-домен (*VLAN Trunking Protocol — протокол магистральных каналов VLAN-сетей*), а также информация о режиме дуплексности между коммутаторами Cisco.
- Протокол CDP может быть отключен глобально или на определенных портах (интерфейсах).

Конфигурация CDP

1. Глобальное включение протокола CDP (*стандартно*).

Система COS	`set cdp enable`
Система IOS	`cdp run` (режим глобальной конфигурации)

Протокол CDP стандартно включен, команды приведены выше. Для его отключения во всем устройстве используется команда `set cdp disable` для COS-устройств или команда `no cdp run` — для устройств IOS.

2. Установка периода обновления для CDP-уведомлений (*необязательно*)

Система COS	`set cdp interval interval`
Система IOS	`cdp timer interval` (режим глобальной конфигурации)

Стандартно уведомления протокола CDP отправляются каждые 60 секунд. Для изменения интервала обновления, указанного в секундах, используются приведенные выше команды. Необходимо принимать во внимание, что интервал обновления должен быть меньше времени удержания информации (holdtime)

3. Установка времени удержания CDP-информации (*необязательно*).

Система COS	`set cdp holdtime interval`
Система IOS	`cdp holdtime interval` (режим глобальной конфигурации)

Если CDP-обновления отсутствуют в течение определенного времени, указанного в параметре `holdtime` в секундах, информация в CDP-таблицы удаляется. Для изменения времени удержания используются указанные выше команды. Время удержания должно быть больше времени отправки уведомлений (как правило, в три раза больше таймера обновления).

4. Настройка отправки параметров CDP-версии для коммутатора (*стандартно*).

Система COS	`set cdp version {v1 | v2}`
Система IOS	`cdp {advertise-v2 | advertise-v1}` (режим глобальной конфигурации)

Существуют две версии протокола CDP — v1 и v2. Эти версии совместимы, однако в версии 2 используется усовершенствованный трехкомпонентный формат сообщений *тип-длина-значение (type-length-values — TLV)*, в котором поддерживаются доменное имя протокола VTP, собственная сеть VLAN и дуплексная информация. Эти сведения чрезвычайно важны для функционирования портов коммутатора. Поступление CDP-сообщений об ошибках свидетельствует о возникновении нефатальных ошибок, однако факт их появления может указывать на возникновение какой-либо проблемы.

5. Отключение протокола CDP на интерфейсе или порту (*необязательно*)

Система COS	`set cdp disable` *mod/port*
Система IOS	`no cdp enable` (режим конфигурирования интерфейса)

Стандартно протокол CDP включен на всех портах. Для портов, которые не подключены к устройствам Cisco, использование механизма CDP не имеет смысла. Для отключения протокола на отдельных портах используются команды, приведенные выше. Для повторного включения протокола в COS-устройствах применяется команда `set cdp enable` *mod/port* и команда `cdp enable` — в IOS-коммутаторах.

С помощью команды `show cdp` в обеих операционных системах отображается общая информация о CDP-конфигурации. Для просмотра информации о соседних устройствах в обеих операционных системах используется команда `show cdp neighbors`. Команды `show cdp interface` *type mod/port* или `show cdp port` *mod/port* отображают сведения по протоколу CDP, касающиеся определенного порта.

Пример конфигурации функции

В этом примере демонстрируется коммутация коммутатора с измененными CDP таймерами: время удержания равно 480 секундам, время обновления — 120 секундам. Кроме того, в примере отключаются порты 3/1-3/48 на COS-коммутаторе и порты Fast Ethernet с 1 по 12 на IOS-коммутаторе.

Ниже приводится пример COS-конфигурации.

```
Console (enable)> set cdp interval 120
Console (enable)> set cdp holdtime 480
Console (enable)> set cdp disable 3/1-48
```

Пример IOS-конфигурации

```
Switch(config)# cdp timer 120
Switch(config)# cdp holdtime 480
Switch(config)# interface fastethernet 0/1
Switch(config)# no cdp enable
Switch(config)# interface fastethernet 0/2
Switch(config)# no cdp enable
Switch(config)# interface fastethernet 0/3
Switch(config)# no cdp enable
Switch(config)# interface fastethernet 0/4
Switch(config)# no cdp enable
Switch(config)# interface fastethernet 0/5
```

```
Switch(config)# no cdp enable
Switch(config)# interface fastethernet 0/6
Switch(config)# no cdp enable
Switch(config)# end
Switch# copy running-config startup-config
```

3.8: установка времени и даты

- Системное время поддерживается программным обеспечением. При инициализации коммутатора системное время устанавливается по аппаратным часам (системному календарю) устройства.
- Поддержка точности системных часов особенно важна, когда необходимо сравнить отображаемую информацию протоколирования и отладочных функций. Коммутатор отмечает эти сообщения временными метками, предоставляя пользователю критерий оценки.
- Системное время поддерживается как *универсальное координированное время* (*coordinated universal time — UTC*), которое также называется *средним временем по гринвичскому меридиану* (*Greenwich Mean Time — GMT*). Формат отображения времени можно настроить с помощью команд операционной системы.
- Системное время может устанавливаться вручную или с помощью *синхронизирующего сетевого протокола* (*Network Time Protocol — NTP*). В дополнение к этому время аппаратных часов коммутатора можно при необходимости корректировать с помощью протокола NTP.
- В протоколе NTP используется понятие *стратум-уровней* (*stratum*) для определения того, насколько близко источник NTP-пакетов (**NTP** speaker) находится к авторитетному источнику времени (атомные или радиоактивные часы). Первый стратум-уровень означает, что **NTP**-сервер непосредственно подключен к авторитетному источнику времени. Протокол **NTP** также сравнивает время, полученное от всех сконфигурированных NTP-узлов, по учитывая данные узлов, имеющих значительные отличия во времени.
- NTP-связь с другими NTP-узлами можно защитить с помощью шифрованной аутентификации.

Третья версия протокола NTP основана на спецификации RFC 1305 и использует UDP-порт 123. Web-сайт www.ccsa.ntel.edu/~ntp/ содержит информацию об открытых NTP-серверах и другие сведения о протоколе.

Внимание!

Коммутаторы Catalyst серий 4000 и 6000, использующие собственную систему IOS, а также коммутаторы 2948G-L3 и 4908G-L3 можно настраивать в качестве авторитетных источников времени для протокола NTP. Сведения по конфигурации этих устройств представлены на странице Web-сайта корпорации Cisco www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fbook_c/fcfprt3/fcf012.htm#41644, а также в книге David Hucaby and Steve McQuerry, *Cisco Field Manual: Router Configuration*, Cisco Press.

Конфигурация функции

Системное время можно устанавливать двумя способами:

- вручную;
- с помощью протокола NTP.

При настройке вручную в устройстве устанавливаются время и дата, а также часовой пояс и необходимость перехода на летнее время. При конфигурировании вручную маршрутизатор/коммутатор не предоставляет способ сохранения установок времени, и возможность проверить точность времени отсутствует. Протокол NTP определяется спецификацией RFC 1305 и предоставляет устройствам в сети механизм получения точного времени от NTP-сервера. При использовании протокола NTP все устройства синхронизируются и поддерживают точное время.

Установка системного времени вручную

1. Установка часового пояса

Система COS	`set timezone` *{zone_name} {hours [minutes]}*
Система IOS	`clock timezone` *zone hrs-offset min-offset* (режим глобальной конфигурации)

В параметре *zone* указывается аббревиатура, соответствующая определенному часовому поясу (EST, PST, CET), которая используется только в целях отображения и может быть назначением любого часового пояса. Фактически отображаемое время определяется с помощью смещения, выраженного в часах *{hrs-offset}* и минутах *{min-offset}* относительно UTC-времени.

2. Конфигурирование перехода на *летнее время (Daylight Savings Time — DST)* (*необязательно*)

Система COS	`set summertime {enable | disable}` *[zone]* `set summertime recurring` *{week day month hh:mm week day month hh:mm} [offset]}* `set summertime date` *month date year hh:mm month date year hh:mm [offset]*
Система IOS	`clock summer-time zone recurring` *{week day month hh:mm week day month hh:mm [offset]}* (режим глобальной конфигурации) `clock summer-time zone date` *{date month | month date} year hh:mm {date month | month date} year hh:mm [offset]* (режим глобальной конфигурации)

В COS-устройствах можно включать и отключать летнее время вручную с помощью команды `set summertime {enable | disable}`. Коммутатор при этом устанавливает время на 60 минут вперед (стандарт США для DST).

Если переход на летнее и зимнее время происходит в определенный день и неделю какого-либо месяца, то данная команда применяется с ключевым словом `recurring`. Для того чтобы запустить или остановить отсчет летнего времени, мож-

но указать номер недели в параметре *week* (включая ключевые слова "first" (первая) и "last" (последний), день недели (*day*) и название месяца (*month*), а также время *hh:mm* в 24-часовом формате. Если команда не содержит аргументов, используется стандарт США с установкой начала отсчета летнего времени в 2:00 в первое воскресенье апреля и завершением в 2:00 в последнее воскресенье октября. Значение параметра *offset* может указываться для установки количества минут, которые добавляются к течению летнего времени (стандартно 60 минут).

Альтернативный способ заключается в использовании ключевого слова **date** для установки точной даты и времени начала и завершения летнего времени в определенном году.

3. Установка (*необязательная*) системных часов (часов IOS).

Система COS	**set time** *[day] [mm/dd/yy] hh:mm:ss*
Система IOS	**clock set** *hh:mm:ss [day month	month day] year* (привилегированный режим)

Часы устанавливаются в момент выполнения этой команды. Время указывается в 24-часовом формате: для IOS-коммутаторов параметр *day* — номер дня месяца, *month* — название месяца, *year* — год в четырехзначном формате. Для COS-коммутаторов дата устанавливается в международном формате.

4. Установка системного календаря (аппаратных часов) (*необязательно*).

Система COS	нет
Система IOS	**calendar set** *hh:mm:ss [day month	month day] year*

Аппаратные часы устанавливаются на определенное время (24-часовой формат) и дату. Параметр *month* — название месяца, *day* — номер дня месяца, *year* — год в полном четырехзначном формате. В качестве альтернативы установить системный календарь можно с системных часов, используя команду EXEC-режима **clock update-calendar**.

Установка системного времени посредством протокола NTP

1. Определение одного или нескольких равноправных NTP-узлов

Система COS	**set ntp server** *ip-addr* [**key** *publickeynum*] **set ntp client enable**
Система IOS	**ntp peer** *ip address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**] (режим глобальной конфигурации)

Равноправный NTP-узел идентифицируется по IP-адресу (*ip address*). Версия протокола NTP (с 1 по 3, стандартная версия — 3) может быть указана с помощью ключевого слова **version**. Если применяется NTP-аутентификация, то используемый ключ аутентификации идентифицируется ключевым словом **key** (конфигурирование функции описано ниже). Если необходимо, можно получить адрес отправителя, используемый в NTP-пакетах по интерфейса, применив ключевое слово **source**. Иначе маршрутизатор использует адрес отправителя на основании адреса исходного интерфейса. Если между равноправными узлами есть

несоответствие, можно использовать ключевое слово **preferred**, которое вынуждает локальный коммутатор синхронизировать время.

2. Конфигурирование широковещательной NTP-службы *(необязательно)*.

Система COS	**set ntp broadcastclient enable** **set ntp broadcastdelay** *microseconds*
Система IOS	**ntp broadcast client** (режим глобальной конфигурации) **ntp broadcastdelay** *microseconds* (режим глобальной конфигурации)

Стандартно сконфигурированные NTP-узлы отправляют и принимают одноадресные пакеты. В случае, если несколько NTP-узлов расположены в общей сети, вместо одноадресных пакетов могут использоваться широковещательные. Отправка широковещательных пакетов включается с помощью команды **ntp broadcast**, а получение — командой **ntp broadcast client**. Команда **ntp broadcastdelay** устанавливает длительность сквозной задержки (round-trip delay) для получения клиентских широковещательных пакетов (от 1 до 999 999 микросекунд, стандартно — 3000 микросекунд).

3. Ограничение доступа к NTP-службе с помощью аутентификации *(необязательно)*

а) Включение NTP-аутентификации

Система COS	**set ntp client enable**
Система IOS	**ntp authenticate** (режим глобальной конфигурации)

б) Указание ключа аутентификации.

Система COS	**set ntp key** *pub/ic-key* **[trusted | untrusted] md5** *secret-key*
Система IOS	**ntp authentication-key** *key-number* **md5** *value* (режим глобальной конфигурации)

Создается ключевое число MD5-аутентификации (*key-number*). Ключу присваивается текстовое значение (*value*) длиной до шести символов открытого текста. После того как конфигурация записывается в NVRAM-память, значение ключа отображается в зашифрованном виде.

в) Применение одного или нескольких ключевых чисел для NTP.

Система COS	**set ntp server** *ip addr* **[key** *public-key***]**
Система IOS	**ntp trusted-key** *key number* (режим глобальной конфигурации)

Удаленным NTP-узлам необходимо аутентифицироваться самостоятельно, используя значение ключевого числа (*key-number*). Для применения к NTP всех необходимых ключей эту команду можно использовать несколько раз.

Пример конфигурирования функции

В примере демонстрируется конфигурация коммутатора, настроенного на восточный часовой пояс США и летнее время. Время устанавливается вручную.

Пример COS-конфигурации.

```
Console (enable)> set timezone EST -5
Console (enable)> set summertime recurring 1 Sunday april 2:00 last Sunday
October 2:00
Console (enable)> set time Saturday 08/11/90 15:30:00
```

Пример IOS-конфигурации.

```
Switch(config)# clock timezone EST -5
Switch(config)# clock summer-time EST recurring 1 sunday april 2:00
last sunday october 2:00
Switch(config)#end
Switch# clock set 15:30:00 August 11 1990
Switch#copy running-config startup-config
```

В приведенной ниже конфигурации протокол NTP включен и настроен на аутентификацию. Ключ **source1key** аутентифицирует узел с адресом 172.17.76.247, а ключ **source2key** — узел с адресом 172.31.31.1.

Ниже приводятся пример COS-конфигурации.

```
Console (enable)> set ntp client enable
Console (enable)> set ntp key 1 trusted md5 source1key1
Console (enable)> set ntp key 2 trusted md5 source1key2
Console (enable)> set ntp server 172.17.76.247 key 1
Console (enable)> set ntp server 172.31.31.1 key 2
```

Пример IOS-конфигурации.

```
Switch(config)# ntp authenticate
Switch(config)# ntp authentication-key 1 md5 sourceA
Switch(config)# ntp authentication-key 2 md5 sourceB
Switch(config)# ntp trusted key 1
Switch(config)# ntp trusted-key 2
Switch(config)# ntp peer 172.17.76.247 key 1
Switch(config)# ntp peer 172.31.31.1 key 2
```

Дополнительная литература

Рекомендуемые ниже источники предоставляют более подробную информацию по темам, рассмотренным в этой главе.

Jim Boyles and David Hucaby, *CCNP Switching Exam Certification Guide*, Cisco Press.

David Hucaby and Steve McQuerry, *Cisco Field Manual: Router Configuration*, Cisco Press.

Michael Wynston, *Cisco Enterprise Management Solutions*, Vol. 1, Cisco Press.

В этой главе...

- **4.1: таблица коммутации.** В этом разделе описываются способы отображения и кеша записей в таблицу коммутации, содержащую MAC-адреса (*Media Access Control — управление доступом к передающей среде*).
- **4.2: выбор порта.** В разделе описываются способы выбора портов коммутатора для конфигурирования.
- **4.3: Ethernet-порты.** В настоящем разделе представлены этапы конфигурирования портов Ethernet, Fast Ethernet, Gigabit Ethernet и 10 Gigabit Ethernet.
- **4.4: порты EtherChannel.** В разделе описываются конфигурационные этапы объединения нескольких портов коммутатора в единый логический канал.
- **4.5: порты Token Ring.** В этом разделе описываются Token Ring-порты коммутатора и методика их конфигурирования.
- **4.6: ATM LANE.** В разделе описывается конфигурирование компонентов *LAN-эмуляции* (LANE), а также мостовое объединение Ethernet-сегментов через ATM-сеть с помощью коммутаторов.

Глава 4

Конфигурация интерфейсов второго уровня

4.1: таблица коммутации

- В таблице коммутации (switching table) содержатся MAC-адреса и порты коммутатора, на которых эти адреса были определены или сконфигурированы статически.
- Пакеты и фреймы перенаправляются с помощью поиска MAC-адреса получателя в таблице коммутации. Фрейм отправляется через соответствующий порт коммутатора.
- Записи таблицы коммутации обычно динамически определяются по мере перемещения пакетов. Записи также могут назначаться статически.

Конфигурация функции

1. Создание статической записи в таблице коммутации (*необязательно*)

Системы COS	`set cam {dynamic | static | permanent} {mac addr | route-descr} mod/ port {vlan-id}`
Системы IOS	`mac-address-table {dynamic | static | secure} mac-addr {vlan vlan-id} {interface int1 {int2 . int15} [protocol {ip | ipx | assigned}]` (режим глобальной конфигурации)

Запись для MAC-адреса получателя *mac-addr* (в COS-коммутаторах в качестве разделителей используются тире, в коммутаторах IOS — точки) создается для того, чтобы указать один или несколько портов коммутатора (*mod/port* в COS-коммутаторах и список интерфейсов для IOS устройств). Если порт получателя является магистральным, необходимо также указать номер VLAN-сети получателя (*vlan-id*).

Записи в таблице коммутации могут иметь тип `static` (статические не устаревают), `dynamic` (динамические записи устаревают; в IOS-коммутаторах Catalyst 6000 недоступны), `permanent` (постоянные записи содержатся в NVRAM-памяти и создаются после перезагрузки коммутатора; в IOS-коммутаторах Catalyst 6000 недоступны) или `secure` (дополнительные записи MAC-адресов могут существовать только на одном порту; в IOS-коммутаторах Catalyst 6000 недоступны).

IOS-коммутатор Catalyst 6000 также связывает определенный MAC-адрес с каким-либо протоколом: `ip` (TCP/IP), `ipx` или `assigned` (или другие протоколы, такие, как DECnet и AppleTalk). Если ключевое слово `protocol` не используется, запись создается для всех трех типов протоколов.

2. Установка времени старения таблицы коммутации (*необязательно*).

Система COS	`set cam agingtime` *vlan-id seconds*
Система IOS	`mac-address-table aging-time` *seconds* `[vlan` *vlan-id*`]` (режим глобальной конфигурации)

Для сети VLAN, номер которой указан в параметре *vlan-id* (COS: 1-1005 и 1025-4094; IOS: 2-1001), записи таблицы коммутации устаревают по истечении времени, указанного в параметре *seconds* (от 0,10 до 1 000 000 секунд, стандартно — 300 секунд). Значение 0 отключает процесс старения информации. В IOS-коммутаторах номер VLAN-сети необязателен. Если номер VLAN не задан, время устаревания устанавливается для всех VLAN-сетей.

3. Удаление записи из таблицы коммутации (*необязательной*).

Система COS	`clear cam` *mac-addr* *[vlan-id]* или `clear cam {dynamic | static | permanent} [` *vlan-id*`]`

Система IOS	`no mac-address-table static` *mac-addr* `[vlan` *vlan-id*`]`
	`[interface` *int1* `[` *int2 ... intn*`]] [protocol {ip | ipx | assigned}]`
	(режим глобальной конфигурации)

Запись можно удалить, ссылаясь на нее по MAC-адресу *mac-addr* (в COS-коммутаторах в качестве разделителей используются тире, в коммутаторах IOS — только). Если запись определена в нескольких VLAN-сетях, необходимо также указать идентификатор *vlan-id*. IOS-коммутаторы позволяют указывать определенные интерфейсы и протоколы.

В COS-коммутаторах можно удалить все записи определенного типа, используя команды `dynamic`, `static` и `permanent`.

4. Определение порта стандартного получателя (*необязательно, только для интерфейсов Catalyst 2900/3500XL*).

Система COS	Нет
Система IOS	`port network` (режим конфигурирования интерфейса)

Коммутаторы, имеющие ограниченный размер таблицы коммутации, в крупной сети не способны определить адреса всех получателей, поэтому можно идентифицировать один интерфейс, который служит в качестве "стандартного" или "сетевого" получателя для какой-либо VLAN-сети.

Если MAC-адрес получателя в таблице коммутации не значится, коммутатор автоматически перенаправляет пакет на такой стандартный сетевой порт вместо лавинной рассылки по всем портам VLAN-сети.

Отображение сведений о таблице коммутации

В табл. 4.1 перечислены команды коммутатора, которые можно использовать для отображения полезной информации о содержимом таблицы коммутации второго уровня.

Таблица 4.1. Команды коммутатора для отображения информации о содержимом таблицы коммутации второго уровня

Функция	Операционная система коммутатора	Команда
Отображает динамически определенные адреса на основании порта или VLAN-номера	COS	`show cam dynamic [{mod/port} | vlan]`
	IOS[1]	`show mac-address-table dynamic [{address mac-addr} | detail | {interface interface interface-number} | {protocol protocol} | {Vlan vlan-id}]`
Отображает статически определенные адреса на основании порта или VLAN-номера	COS	`show cam {static | permanent | system} [{mod/port} | vlan]`
	IOS	`show mac-address-table static [address mac-addr | detail | interface interface interface-number | protocol protocol | Vlan vlan-id]`
Отображает порт или VLAN-сеть, связанную с какими-либо MAC-адресом	COS	`show cam mac_addr [vlan]`
	IOS	`show mac-address-table address mac-addr [detail | {interface interface interface-number} | {protocol protocol} | {Vlan vlan-id} | all]`
Отображает время старения таблицы коммутации	COS	`show cam agingtime [vlan]`
	IOS	`show mac-address-table aging-time [Vlan vlan-id]`
Отображает счетчик адресов и размер таблицы коммутации	COS	`show cam count {dynamic | static | permanent | system} [vlan]`
	IOS	`show mac-address-table count [Vlan vlan-id] [slot slot-num]`

[1] *В устройствах, работающих под управлением операционной системы IOS, все команды, которые указаны в таблице, вводятся в режиме привилегированного пользователя. — Прим. ред.*

Пример таблицы коммутации

Предположим, необходимо установить расположение порта коммутатора, к которому подключен определенный персональный компьютер. MAC-адрес компьютера — 00-b0-d0-f5-45-0e.

Система COS	`show cam 00-b0-d0-f5-45-0e`
Система IOS	`show mac-address-table address 00b0.d0f5.450e` (в режиме привилегированного пользователя)

Ниже приводится пример информации, отображаемой COS-коммутатором.

```
switch-cos (enable) show cam 00-b0-d0-f5-45-0e
* - Static Entry. + - Permanent Entry. # - System Entry. R - Router Entry.
X - Port Security Entry $ - Dotlx Security Entry
VLAN Dest MAC/Route Des [CoS] Destination Ports or VCs / [Protocol Type]
---- ----------------- ----- -------------------------------------------
536 00-b0-d0-f5-45-0e 1/2 [ALL]
Total Matching CAM Entries Displayed =1
```

Пример информации, полученной от IOS-коммутатора.

```
switch-ios#show mac-address-table address 00d0.b7a5.4dc1
Non static Address Table:
Destination Address Address Type VLAN Destination Port
------------------- ------------ ---- --------------------
00d0.b7a5.4dc1 Dynamic 534 FastEthernet0/2
```

Предположим, необходимо получить список всех MAC-адресов, определенных на указанном порту коммутатора.

Система COS	`show cam dynamic 3/1`
Системы IOS	`show mac-address-table dynamic interface gigabit 0/1` (в режиме привилегированного пользователя)

COS-коммутатор отображает следующие сведения:

```
cos-switch (enable) show cam dynamic 3/1
* - Static Entry. + - Permanent Entry. # - System Entry. R - Router Entry.
X - Port Security Entry $ - Dotlx Security Entry
VLAN Dest MAC/Route Des [CoS] Destination Ports or VCs / [Protocol Type]
---- ----------------- ----- -------------------------------------------
959 00-00-0c-45-21-00 3/1 [ALL]
64 00 00 1b 04 2f 76 3/1 [ALL]
57 00-00 48 0a-3b-0b 3/1 [ALL]
Total Matching CAM Entries Displayed =1
```

Ниже представлена информация, отображаемая IOS-коммутатором.

```
switch-ios#show mac-address-table dynamic interface gig 0/1
Non-static Address Table:
Destination Address Address Type VLAN Destination Port
------------------- ------------ ---- --------------------
0500.0c45.2100 Dynamic 959 GigabitEthernet0/1
0500.1b04.2f76 Dynamic 64 GigabitEthernet0/1
0003.485a.3b0b Dynamic 57 GigabitEthernet0/1
```

Совет

Если необходимо определить расположение какого-либо MAC-адреса внутри крупной сети и неизвестно, откуда следует начать поиск, рекомендуется начать с коммутатора основного уровня, расположенного ближе к центру сети. Искать необходимый MAC-адрес следует в таблице коммутации. Когда адрес будет найден, нужно перейти к соседнему коммутатору, подключенному к данному порту получателя.

Далее рекомендуется продолжать поиск адреса в таблицах коммутации, последовательно перемещаясь по соседним коммутаторам до тех пор, пока не будет достигнута граница сети, к которой физически подключено данное устройство.

4.2: выбор порта

- При конфигурировании порта или интерфейса второго уровня порт прежде всего необходимо выбрать или идентифицировать.
- COS-коммутатор допускает ввод в одной командной строке одного порта, списка портов, диапазона портов или их комбинации.
- IOS-коммутаторы позволяют указывать в одной командной строке только один интерфейс[1].
- IOS-коммутаторы Catalyst 6000 позволяют определить диапазон интерфейсов. Любые последующие команды конфигурирования интерфейсов применяются к данному диапазону интерфейсов.

Конфигурация функции

1. Выбор порта

Система COS	`set` .. *mod/port* ..
Система IOS	`interface` *type mod/num* (режим глобальной конфигурации)

Порт COS-коммутатора идентифицируется по номерам модуля (*mod*) и порта (*port*). Порт IOS-коммутатора называется интерфейсом и идентифицируется по типу (`fastethernet`, `gigabitethernet` и т.п.), номеру модуля (*mod*) и номеру порта (*num*).

Совет

Нередко для нескольких портов коммутатора вносятся одинаковые конфигурационные изменения. В COS-коммутаторах в одной команде можно просто сослаться на несколько портов (см. следующий этап). В то же время IOS-коммутаторы позволяют одновременно конфигурировать только один интерфейс, что делает весьма утомительной процедуру внесения изменений VLAN-доступа или скорости порта на всех 48 портах коммутатора Catalyst 3500XL.

[1] *В некоторых новых моделях коммутаторов Catalyst под управлением расширенной операционной системы IOS версии 12.x можно указывать несколько интерфейсов в одной командной строке с определенным режимом конфигурирования. — Прим. ред.*

Вместо того чтобы индивидуально вводить конфигурацию для каждого из 48 портов, рекомендуется использовать текстовый редактор персонального компьютера для создания конфигурационного шаблона. Можно копировать команды и использовать функции глобального изменения или замены для изменения большого количества параметров, после чего данный файл следует скопировать в эмулятор терминала или загрузить его в коммутатор по протоколу TFTP.

Альтернативой является использование web-интерфейса в кластере коммутаторов. С помощью графических средств можно быстро внести индивидуальные изменения конфигурации портов.

2. Выбор диапазона портов.

Система COS	`set ... mod/port ...`
Система IOS	**`interface range`** *`port-range`* или **`define interface-range`** *`macro-name port-range`* **`interface range macro`** *`macro-name`* (все команды вводятся в режиме глобальной конфигурации)

COS-коммутаторы позволяют указывать множество портов, разделенных запятыми (без пробелов). Номера портов на одном модуле можно указывать, разделяя их дефисом. Например, модуль 4 порт 1 и модуль 5 порт 1 обозначаются как `4/1,5/1`, тогда как порты с 1 по 4 на модулях 4 и 5 наряду с портом 3 модуля 6 указываются как `4/1-4,5/1-4,6/3`. Вводить диапазон или список портов можно всякий раз, когда в какой-либо COS-команде необходимо указать номера модуля и порта.

IOS-коммутатор Catalyst 6000 позволяет вводить списки или диапазоны интерфейсов один раз, с тем чтобы последующие команды применялись к каждому из указанных интерфейсов. Диапазон портов *`port-range`* определяется как тип интерфейса (**`fastethernet`**, **`gigabitethernet`**, **`tengigabitethernet`** или **`vlan`**), за которым следует номер модуля, косая черта (/) и начальный номер порта. Конец диапазона указывается с помощью двух пробелов, разделенных дефисом, и конечного номера порта. Если указывается несколько диапазонов, то их необходимо отделять друг от друга с помощью запятой.

Основной формат диапазона — *`type slot/first-port - last-port [,type slot/first-port - last-port ...]`* (т.е. формат вида "тип гнездо/первый порт — последний порт"), можно перечислить до пяти различных диапазонов. После ввода команды **`interface range`** система переходит в режим конфигурирования интерфейса.

Если к диапазону интерфейсов требуется применить ряд конфигурационных изменений, можно создать макрос, содержащий список данных диапазонов. Для этого используют команду **`define interface-range`** с указанием имени макроса (произвольное текстовое имя) в параметре *`macro-name`* и диапазона портов в параметре *`port-range`* (списка диапазонов, определенного ранее). Макрос может быть сохранен в конфигурации коммутатора для последующего использования. Для запуска макроса диапазона используется команда **`interface range macro`** с параметром *`macro-name`* (имя макроса).

Пример выбора портов

Необходимо установить режим автоматического согласования скорости портов 1 и 2 модуля 1, а также портов 1-4 модуля 6. (Может использоваться любая конфигурационная функция, скорость портов выбрана в данном случае только в целях демонстрации выбора портов.) COS-коммутатор позволяет конфигурировать все порты в одной команде. В случае IOS-коммутатора (Catalyst 2900/3500XL) необходимо конфигурировать каждый интерфейс отдельно. Наконец, IOS-коммутатор Catalyst 6000 позволяет идентифицировать указанные порты в виде двух диапазонов и установить их скорости с помощью одной команды конфигурирования интерфейса.

Система COS	`set port speed 1/1-2,6/1-4 auto`

Система IOS	`interface gig 1/1`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)
	`interface gig 6`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)
	`interface gig 6/1`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)
	`interface gig 6/2`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)
	`interface gig 6/3`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)
	`interface gig 6/4`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)

Catalyst 6000 IOS	`interface range gigabitethernet 1/1 - 2,`
	`gigabitethernet 6/1 - 4`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)
	или
	`define interface-range AnnexPorts gigabitethernet 1/1`
	`- 2, gigabitethernet 6/1 - 4`
	`interface range macro AnnexPorts`
	(режим глобальной конфигурации)
	`speed auto`
	(режим конфигурирования интерфейса)

4.3: Ethernet-порты

- Автоматическое согласование скорости канала и режима дуплексности для портов 10/100/1000BASE-T возможно с помощью функций, стандартизированных в спецификациях IEEE 802.3u и 802.3ab. Две конечные точки соединения обмениваются сведениями о допустимых параметрах передачи и выбирают наивысшую общую скорость и дуплекс, поддерживаемые ими обеими.
- Ethernet-порты обозначаются с помощью номеров модуля и порта (*mod/port*) на COS-коммутаторах и с помощью типа и номера интерфейса (`interface` и одно из ключевых слов `ethernet`, `fastethernet`, `gigabitethernet` или `tengigabitethernet`) на IOS-коммутаторах.
- Если в порту обнаруживаются определенные проблемы, коммутатор автоматически переводит его в состояние *errDisable*, или "отключен из-за ошибки" (error disable). В результате влияние проблемного порта на остальную часть сети сводится к минимуму.
- Порты, находящиеся в состоянии *errDisable*, могут быть автоматически включены или восстановлены по истечении периода ожидания или вручную. В любом случае прежде чем пытаться восстановить errDisable-порты, следует определить и исправить условия, вызывающие проблему.

Конфигурирование портов

1. Присвоение порту описательного имени (*необязательно*).

Система COS	`set port name` *mod/port* [*port-name*]
Система IOS	`description` *port-name* (режим конфигурирования интерфейса)

Описание (*port-name*) — строка текста, которая назначается порту для облегчения работы пользователя. Как правило, в описание включается ссылка на местоположение, функцию или пользователя порта.

2. Установка скорости порта (*необязательно*).

Система COS	`set port speed` *mod/port* {`10` | `100` | `1000` | `auto`}
Система IOS	`speed` {`10` | `100` | `1000` | `auto` | `nonegotiate`} (режим конфигурирования интерфейса)

Существуют следующие значения скорости порта: `10` (10 Мбит/с для портов 10, 10/100 и 10/100/1000BASE-T), `100` (100 Мбит/с для портов 10/100 и 10/100/1000BASE-T), `auto` (автоматическое согласование скорости для портов 10/100 и 10/100/1000BASE-T; стандартное значение) и `nonegotiate` (не согласовывать скорость; только для IOS-коммутаторов). Скорости портов 10BASE-T, 100FX и *GBIC (Gigabit Interface Converter — конвертер гигабитного интерфейса)* являются фиксированными и не могут устанавливаться с помощью данной команды.

Совет

Выбор скорости `auto` для порта (стандартная настройка) позволяет ему *участвовать в согласовании* с дальним концом канала. Две конечные точки обмениваются информаци-

ней о возможных параметрах передачи и выбирают наилучшую скорость и режим дуплексности, поддерживаемые ими обеими. Если в одной из точек автоматическое согласование отключено, то другая точка может определить только скорость канала, исходя из параметров электрических сигналов. Режим дуплекса в таком случае не может быть определен и остается стандартным.

Если требуется настроить скорость и режим дуплекса порта коммутатора на значение, отличное от `auto`, то такие же значения следует установить в устройстве на дальнем конце канала.

Как правило, если порт 10/100/1000BASE-T коммутатора подключен к подобному порту другого коммутатора или к критически важному устройству, такому, как сервер, маршрутизатор или межсетевой экран, то наилучшим будет установка скорости и режима дуплексности в фиксированное значение. Такие настройки устраняют любую возможность автоматического согласования, в дальнейшем вынуждающую порт понимать скорость.

3. Установка режима дуплексности порта (*необязательно*).

Система COS	`set port duplex` *mod/port* `{full	half	auto}`
Система IOS	`duplex {full	half	auto}` (режим конфигурирования интерфейса)

Режим дуплексности можно установить в одно из следующих значений: `full` (дуплекс), `half` (полудуплекс) (стандартная настройка IOS) или `auto` (автоматическое согласование режима дуплексности; стандартно для COS). Параметр `auto` недоступен для IOS-коммутаторов Catalyst 6000, если скорость установлена в `auto`, то выбирается подходящий дуплекс.

Совет

Режим дуплексности может быть согласован автоматически, только если скорость порта также установлена в `auto` (автоматическое согласование). Порты Gigabit и 10 Gigabit Ethernet могут устанавливаться либо в дуплексный, либо в полудуплексный режим. Следует избегать использования портов, имеющих ошибки дуплексности, в которых один конец работает в дуплексном режиме, а второй — в полудуплексном. При таких условиях возможен слабый отклик и высокий уровень ошибок. Необходимо убедиться, что оба конца канала установлены в режим автоматического согласования или имеют одинаковые режимы дуплексности.

4. Настройка управления потоком данных в порту (*необязательно*).

Система COS	`set port flowcontrol` *mod/port* `{receive	send} {off	on	desired}`
Система IOS	`flowcontrol {send	receive} {desired	off	on}` (режим конфигурирования интерфейса)

Порт коммутатора может получать фреймы *паузы (pause)*, на короткое время приостанавливающие передачу при заполнении буферов на дальнем конце. Стандартно обработка получения (`receive`) отключена (`off`) для всех типов портов коммутатора (кроме 10 Gigabit Ethernet). Порт также может отправлять фреймы паузы, если его буферы заполнены. Стандартно для портов Fast Ethernet передача (`send`) включена (`on`), предпочтительна (`desired`) для Gigabit и выключена (`off`) для всех остальных типов портов. Использование ключевого слова `de-`

`mixed` возможно только для портов Gigabit, в которых автоматическое согласование является неотъемлемой функцией.

5. Управление согласованием порта (*необязательно*)

Система COS	`set port negotiation` *mod/port* `{enable | disable}`
Система IOS	`[no] negotiation auto` (режим конфигурирования интерфейса)

Стандартно на Gigabit Ethernet-портах согласование канала (управление потоком, дуплексность, сведения об отказах) включено. Для отключения этой функции используется ключевое слово `disable` или `no`.

6. Включение таймера отбрасывания (debounce timer) (*необязательно, только для устройств Catalyst 6000*).

Система COS	`set port debounce` *mod/port* `{enable | disable}`
Система IOS	Нет

Стандартно линейные платы ожидают 300 миллисекунд (10 миллисекунд для волоконно-оптических Gigabit-портов), прежде чем объявить главному процессору об изменении состояния порта. Таким образом "сдерживается" отправка уведомлений об изменении состояния (активно/неактивно). В результате при быстрой смене состояния не происходят прерывания STP (*Spanning Tree Protocol — протокол распределенного связующего дерева*), PAgP (*Port Aggregation Protocol — протокол агрегирования портов*), SNMP (*Simple Network Management Protocol — простой протокол управления сетью*) и других. "Сдерживание" предоставляет порту возможность сохранять стабильное состояние. Если выяснится, что длительность данного периода недостаточна, то можно включить (`enable`) расширенное "сдерживание" для определенных портов, при котором период "сдерживания" равен 3,1 секунды (100 миллисекунд для волоконно-оптических Gigabit-портов).

Внимание!

Когда данная функция включена, обнаружение изменений состояния порта задерживается. Обычная STP-обработка состояния наряду с PAgP-согласованием может вызвать весьма большую задержку, прежде чем порт можно будет использовать. Рекомендуется с осторожностью применять описанную функцию.

7. Оптимизация порта как соединения с одним узлом (*необязательно*).

Система COS	`set port host` *mod/port*

Система IOS	`spanning-tree portfast`
	`switchport mode access`
	`no channel-group`
	(все команды вводятся в режиме конфигурирования интерфейса)

Для порта устанавливаются несколько параметров: включена функция STP PortFast, режим магистрального канала отключен, EtherChannel-канал отключен, использование dot1q-туннелей запрещено. Такие настройки оптимизируют время активизации канала при подключении порта только к одному узлу. COS-

коммутаторы способны реализовать указанные настройки в одной команде, тогда как IOS-коммутаторам требуется несколько командных строк.

8. Использование линейного питания при подключении IP-телефона (*необязательно*).

Система COS	`set port inlinepower` *mod/port* `{off | auto}`
Система IOS	`power inline {auto | never}` (режим конфигурирования интерфейса)

В портах или линейных платах, которые способны поддерживать линейное питание (inline power) для IP-телефонов, стандартно (`auto`) питание подается, если на данном порту обнаружен IP-телефон. Если питание к подключенному устройству не следует подавать вообще, выбирается ключевое слово `off` или `never`. Более подробные сведения по конфигурированию приведены в главе 14. "Поддержка передачи голосовых данных".

9. Использование больших, или *jumbo*-фреймов (*необязательно*).

Система COS	`set port jumbo` *mod/port* `{enable | disable}`
Система IOS	`mtu` *bytes* (режим конфигурирования интерфейса)

Стандартно максимальный размер фрейма, или *максимально поддержку блок передачи данных (Maximum Transmission Unit — MTU)*, который может быть обработан коммутатором, равен 1548 байтам (COS; стандартно отключено (`disable`)) и 1500 байтам (IOS; `mtu 1500`). Иногда возникает необходимость в коммутации более крупных пакетов для увеличения производительности передачи данных от сервера к серверу. Для того чтобы разрешить коммутацию пакетов размером до 9216 байтов, используется ключевое слово `enable` (COS) или устанавливается MTU-размер в *байтах* (IOS; от 1500 до 9216).

Совет

Включение поддержки jumbo-фреймов позволяет коммутировать большие фреймы. Если их также необходимо маршрутизировать, следует обеспечить такие же значения MTU на соответствующих интерфейсах маршрутизатора. Поддержка jumbo-фреймов доступна на плате MSFC2 и включается с помощью команды `mtu`, но недоступна на функциональной плате многоуровневой коммутации (Multilayer Switch Feature Card — MSFC).

10. Автоматическое восстановление портов из состояния errDisable (*необязательно*).

    а) Установка периода ожидания перед автоматическим восстановлением портов

Система COS	`set errdisable-timeout interval` *{interval}*
Система IOS	`errdisable recovery {interval` *interval}* (режим глобальной конфигурации)

При автоматическом выходе из состояния errDisable порты остаются отключенными на протяжении определенного интервала (`interval`) времени (от 30 до 86400 секунд, стандартно 300 секунд). Эта команда недоступна для коммутаторов семейств Catalyst 2900XL и IOS-коммутаторов 3500XL.

б) Указание причины для автоматического восстановления портов

Система COS	`set errdisable-timeout {enable | disable} reason`
Система IOS	[no] `errdisable recovery cause reason` (режим глобальной конфигурации)

Стандартно порты в состоянии errDisable автоматически не восстанавливаются и не включаются. Если требуется автоматическое восстановление для errDisable-портов, используется ключевое слово **enable** (COS-коммутаторы) или команда **errdisable recovery cause** (IOS-коммутаторы). Порты восстанавливаются по истечении периода ожидания (errDisable timeout). Но перед этим нужно выбрать одну из описанных ниже причин восстановления.

- **BPDU Port Guard** (контроль BPDU-сообщений) — *блок данных протокола моста* (*bridge protocol data unit — BPDU*) принимается на порту в состоянии STP PortFast; используется причина (параметр *reason*), обозначаемая ключевым словом **bpdu-guard** (для COS), или **bpduguard** (для IOS)
- **UDLD** (unidirectional link — однонаправленность канала) — обнаружен однонаправленный канал, используется причина, обозначаемая ключевым словом **udld** (для IOS и COS).
- **STP Root Guard** (контроль корневого устройства STP) — используется причина, обозначаемая ключевым словом **rootguard** (только в IOS).
- **Ошибка конфигурирования EtherChannel-канала** —отсутствует согласованная конфигурация портов EtherChannel, используется причина, обозначаемая ключевым словом **channel-misconfig** (COS) или **pagp-flap** (IOS).
- **Самопроизвольное изменение магистрального согласования** (link negotiation flapping) — *динамический магистральный протокол* (*Dynamic Trunking Protocol — DTP*) обнаруживает изменения магистральной инкапсуляции; используется причина, обозначаемая ключевым словом **dtp-flap** (только в IOS)
- **Ошибка дуплексности** - обнаруживается множество чрезмерных и поздних коллизий; используется причина, обозначаемая ключевым словом **duplex-mismatch** (только COS).
- **Порты самопроизвольно отключаются и включаются** — используется причина, обозначаемая ключевым словом **link-flap** (только IOS).
- **Некоторые другие проблемы** — проблемы, обнаруженные коммутатором, не включенные в данный список; используется причина, обозначаемая ключевым словом **other** (только COS).
- **Все известные причины перехода в состояние errDisable** — порты переходят в состояние errDisable при обнаружении какой-либо из перечисленных проблем, используется причина, обозначаемая ключевым словом **all** (COS и IOS).

В коммутаторе Catalyst 2900XL и IOS-коммутаторе семейства 3500XL команда **errdisable recovery cause** недоступна.

11. Отключение или включение порта

Система COS	`set port enable` *mod/port* или `set port disable` *mod/port*

Система IOS	`shutdown`
	или
	`no shutdown`
	(обе команды вводятся в режиме конфигурирования интерфейса)

Стандартно порт является включенным (т.е. введена команда `enable` или `no shutdown`). Для отключения порта используется ключевое слово `disable` или `shutdown`.

Пример конфигурации Ethernet-порта

Предположим, 10/100/1000-порт коммутатора подключен к почтовому серверу. Порт настроен для работы со скоростью 100 Мбит/с в дуплексном режиме. Кроме того, порт настроен на один узел, поэтому задержка при включении, связанная с протоколами PAgP, STP или согласованием магистральных каналов, отсутствует.

Система COS	`set port name 3/1 Mail server`
	`set port speed 3/1 100`
	`set port duplex 3/1 full`
	`set port host 3/1`
	`set port enable 3/1`

Система IOS	`description Mail server`
	`speed 100`
	`duplex full`
	`spanning-tree portfast`
	`switchport mode access`
	`no channel-group`
	`no shutdown`
	(все команды вводятся в режиме конфигурирования интерфейса)

Отображение сведений об интерфейсах второго уровня

В табл. 4.2 перечислены наиболее полезные команды коммутатора, которые можно использовать для отображения полезной информации об интерфейсах второго уровня.

Таблица 4.2. Команды коммутатора для отображения сведений об интерфейсах второго уровня

Функция отображения	Операционная система коммутатора	Команда
Состояние порта	COS	`show port [mod	mod/port]`
	IOS	`show interfaces [type num]`
Счетчик ошибок порта	COS	`show port mac [mod	mod/port]`

Функция отображения	Операционная система коммутатора	Команда
	IOS	`show interfaces counters [broadcast | errors | {module mod-num} | {trunk [module mod-num]}]`
MAC-адрес порта, используемый коммутатором	COS	`show port mac-address [mod [/port]]` или `show module mod`
	IOS	`show interfaces [type num]` или `show catalyst6000 chassis-mac-address`
Управление потоком в порту	COS	`show port flowcontrol [mod [/port]]`
	IOS	`show interfaces [interface [mod]] flow-control`
Согласование портов	COS	`show port negotiation [mod [/port]]`
	IOS	Нет
Сдерживание портов	COS	`show port debounce [mod [ mod/port]]`
	IOS	Нет
Линейное питание порта	COS	`show port inlinepower [mod [/port]]`
	IOS	`show power inline [interface-id] [actual | configured]`
Поддержка jumbo-фреймов	COS	`show port jumbo`
	IOS	`show interfaces [type num]`
errDisable — восстановление и состояние порта	COS	`show errdisable-timeout`
	IOS	`show errdisable recovery`

COS-коммутаторы позволяют создавать и просматривать отчеты об использовании устройства, объемах трафика и ошибках в каждом порту коммутатора. Эти так называемые *TopN-отчеты* могут оказаться полезными при отсутствии приложений управления сетью, которые генерируют статистические отчеты о портах коммутатора.

1. Запуск TopN-отчета.

Система COS	`show top [N] [metric] [interval interval] [port-type] background`
Система IOS	Нет

Отчет должен включать в себя верхние N (от 1 до максимального количества портов коммутатора; стандартно 20 портов) записей, используя параметр `metric` в качестве ключа сортировки. Параметр `metric` может принимать одно из следующих значений: `util` (использование; стандартное значение), `bytes` (байтов на входе/выходе), `pkts` (пакетов на входе/выходе), `bcst` (широковещательных пакетов на входе/выходе), `mcst` (многоадресатных пакетов на входе/выходе), `errors` (входных ошибок) или `overflow` (переполнение буфера).

TopN-отчет может проверять данные порта через определенный интервал времени (`interval`) (от 1 или 10 до 999 секунд; стандартно 30 секунд). Если параметр `interval` равен нулю, то в отчете используются абсолютные счётчики портов (которые отображаются с помощью команд `show port` или `show port mac`). В дополнение к этому отчеты могут создаваться для определенных типов портов (`port-type`), которые указываются с помощью следующих ключевых слов: `all` (стандартно все типы портов), `eth` (все типы Ethernet-портов), `10e` (10 Мбит/с Ethernet), `fe` (Fast Ethernet), `ge` (Gigabit Ethernet) или `10ge` (10 Gigabit Ethernet).

Коммутатор выводит сообщение о начале и завершении создания отчета. Следует зафиксировать номер генерируемого отчета. С помощью ключевого слова `background` (фоновый режим) в процессе обработки отчета можно вводить другие команды коммутатора.

2. Просмотр сохраненного TopN-отчета.

Система COS	`show top report [report-num]`
Система IOS	Нет

С помощью параметра `report-num` можно просмотреть определенный TopN-отчет. Для просмотра списка всех сохраненных отчетов это параметр следует пропустить. Отчеты хранятся в памяти коммутатора и остаются там до перезагрузки или отключения питания. Для очистки записей TopN-отчетов используется команда `clear top {all | report-num}`.

4.4: порты EtherChannel

- Можно агрегировать несколько отдельных портов коммутатора в один логический порт или порт EtherChannel.
- При объединении портов Fast Ethernet формируются порт *Fast EtherChannel (FEC)*, соответственно объединение гигабитовых портов образует порт *Gigabit EtherChannel (GEC)*.
- Можно конфигурировать порты EtherChannel вручную или агрегировать их с помощью динамических протоколов. Протокол PAgP представляет собой фирменное решение корпорации Cisco, тогда как LACP (*Link Aggregation Control Protocol — протокол управления агрегированием каналов*) — стандартизированный протокол, определенный в спецификации IEEE 802.3ad (которая также называется стандартом IEEE 802.3, статья 43, "Агрегирование каналов" (IEEE 802.3 Clause 43, "Link Aggregation")).
- Фреймы распределяются по отдельным портам, составляющим EtherChannel, с помощью алгоритма хеширования. В алгоритме могут использоваться IP-адреса

отправителя, получателя, комбинации их адресов. MAC-адреса отправителя и получателя или номера портов TCP/UDP в зависимости от аппаратной платформы и конфигурации.

- Распределение фреймов предопределено, т.е. одна и та же комбинация адресов или номеров портов всегда указывает на один и тот же порт внутри EtherChannel.
- Для выбора канала, по которому будет передан фрейм, в хеш-алгоритме распределения фреймов используется XOR-операция (eXclusive-OR (XOR) — исключающего "или") для одного или нескольких младших битов адресов или номеров портов TCP/UDP. Для двухпортового банка портов используется последний бит, для четырехпортового — два последних бита и для восьмипортового — три последних. (При выполнении XOR-операции если два бита являются идентичными, результат равен нулю, а если два бита различны, результат равен единице.)
- Если канал внутри EtherChannel-канала выходит из строя, то трафик, следовавший по отказавшему каналу в нормальном режиме, перемещается в оставшиеся каналы.
- EtherChannel-каналы могут быть статическими портами доступа или магистральными портами. Однако прежде чем формировать EtherChannel, всем объединяемым каналам необходимо задать согласованную конфигурацию.

Внимание!

Протокол PAgP отправляет фреймы на адрес получателя 01-00-0C-CC-CC-CC как протокол доступа к подсети (802.2 Subnetwork Access Protocol — SNAP) 0x000C0104. Протокол LACP отправляет фреймы на адрес получателя 01-80-c2-00-00-02, используя протокол 0x8809.

Конфигурация функции

1. Выбор EtherChannel-протокола для модуля (*необязательно; только для COS-коммутаторов Catalyst 6000*).

Система COS	`set channelprotocol {pagp | lacp} mod`
Система IOS	Нет

Для динамического управления группой EtherChannel по всех модулях транспортным является протокол PAgP (pagp). Если нужно, для модуля с номером mod можно выбрать протокол LACP, используя ключевое слово lacp.

Совет

Протоколы PAgP и LACP несовместимы. Следовательно, в модулях и портах на обоих концах потенциального EtherChannel-канала следует использовать один и тот же протокол.

2. Настройка STP-стоимости для EtherChannel-порта (*необязательно*).

   а) Установка STP-стоимости порта.

Система COS	`set spantree channelcost {channel-id | all} cost`
Система IOS	Нет

Стандартно STP-стоимость порта для EtherChannel-канала основывается на стоимости порта агрегированной полосы пропускания. Например, стоимость одного порта на 100 Мбит/с равна 19-ти. Если в банке объединяются два 100 Мбит/с порта в качестве FEC, то стоимость порта для 200 Мбит/с равна 12-ти. В банке из четырех 100 Мбит/с портов стоимость равна 8-ми для 400 Мбит/с. Значения STP-стоимости портов приведены в табл. 7.1 главы 7. "Протокол распределенного связующего дерева (STP)".

Стоимость порта для всех EtherChannel-каналов можно изменить, используя ключевое слово **all**. Для изменения стоимости порта одного EtherChannel-канала необходимо указать его номер (*channel-id*). Чтобы определить этот номер, используется команда **show channel group** (протокол PAgP) или **show lacp-channel group** (протокол LACP). Идентификатор *channel-id* представляет собой уникальный номер, автоматически назначаемый EtherChannel-каналу.

STP-стоимость порта определяется в параметре *cost* (1-65535 в 16-битовом "коротком формате" или 1-4294967296 в 32-битовом "длинном формате"). Более подробные сведения по стоимости приведены в разделе "7.1: принцип действия протокола STP".

б) Установка STP-стоимости порта для каждой VLAN-сети.

Система COS	**set spantree channelvlancost** *channel-id cost* **set spantree portvlancost** *mod/port* [**cost** *cost*] [*vlan-list*]
Система IOS	Нет

Для того чтобы разрешить конфигурирование стоимости портов в каждой сети VLAN для EtherChannel с номером *channel-id*, используется команда **set spantree channelvlancost**. STP-стоимость портов устанавливается равной значению параметра *cost* для всех VLAN-сетей, которые будут поддерживаться данным EtherChannel-каналом. После чего следует задать стоимость порта для определенных VLAN-сетей с помощью команды **set spantree portvlancost**. Более подробные сведения по стоимости приведены в разделе "7.1: принцип действия протокола STP".

3. Использование протокола PAgP в EtherChannel-порту (*необязательно*)

Совет

При внесении конфигурационных изменений для добавления или удаления портов из EtherChannel-порта необходимо учитывать их влияние на распределенное связующее дерево, что особенно важно в реально функционирующей сети, где изменения могут привести к нарушению работы служб.

Протокол STP функционирует на EtherChannel как на обычном порту коммутатора. После того как в EtherChannel были добавлены порты, протокол STP проходит через различные состояния, чтобы обеспечить бесперебойную топологию. Порты коммутатора, входящие в административную группу EtherChannel, можно включать или отключать, не вызывая изменений STP-топологии. В результате остальные каналы в EtherChannel остаются в "передающем" STP-состоянии.

Однако при попытке добавить новый порт в активную административную группу EtherChannel запускается изменение STP-топологии. То же происходит при изменении номера административной группы на активном EtherChannel-канале. В таком случае появляется переконфигурированный логический канал, поэтому STP вводит

EtherChannel (и все входящие в него порты) в состояние "прослушивания" (listening) и "изучения топологии" (learning), что прерывает трафик группы EtherChannel на период длительностью до 50 секунд.

а) Назначение портов EtherChannel-каналу.

Система COS	`set port channel` *mod/port* *(admin group)*
Система IOS	Нет

В качестве EtherChannel-порта назначаются один или несколько портов, определенных в параметре *mod/port*. Если необходимо, можно указать номер определенной группы (*admin group*). В случае, если данный параметр пропущен, коммутатор автоматически присваивает портам новый уникальный номер группы. Если номер группы указан и уже используется, новый EtherChannel-порт получает данный номер группы, а порты, входившие в нее ранее, получают другой уникальный номер группы.

В IOS-коммутаторах порты назначаются EtherChannel-группе в момент установки режима протокола PAgP. Эта операция осуществляется на этапе, который описан ниже.

Совет

Все порты, которые будут принадлежать группе EtherChannel, необходимо перечислить в одной команде. Для того чтобы добавить или удалить отдельные порты из банка, необходимо повторно ввести данную команду с обновленным списком всех необходимых портов.

б) Установка режима протокола PAgP.

Система COS	`set port channel` *mod/port* `mode {on | off | desirable | auto} {silent | non-silent}`
Система IOS	`channel-group` *number* `mode {on | auto [non-silent] | desirable [non-silent]}` (режим конфигурирования интерфейса)

Обращение к каналу производится по одному из его портов в параметре *mod/port* (COS) или путем выбора интерфейса и номера группы (*number*) (IOS). Настроить протокол PAgP можно в одном из следующих режимов: `on` (включен; EtherChannel-канал используется, но PAgP-пакеты не отправляются), `off` (выключен; EtherChannel-порт отключен), `desirable` (желаемый режим работы; коммутатор готов к активному формированию группы EtherChannel; PAgP-пакеты отправляются) или `auto` (коммутатор готов к пассивному формированию группы EtherChannel, PAgP-пакеты не отправляются; стандартный режим).

В режимах `auto` и `desirable`, прежде чем появится возможность согласовать и сформировать канал EtherChannel, требуется отправка PAgP-пакетов. Однако бывают случаи, когда один конец EtherChannel-канала (сервер или сетевой анализатор) не генерирует PAgP-пакеты или находится в состоянии паузы. Можно использовать ключевое слово `silent` (стандартная установка) для того, чтобы разрешить переход порта в режим EtherChannel с агрегатом в состоянии паузы после пятнадцатисекундной задержки. Чтобы сделать обяза-

тельным PAgP-согласованием перед активизацией EtherChannel-канала, используется ключевое слово `non-silent`.

Совет

Протокол PAgP недоступен на платформах Catalyst 2900 и 3500XL. EtherChannel-порт на этих коммутаторах работает либо в режиме `on`, либо в режиме `off` без согласования. При подключении EtherChannel-канала от другой коммутирующей платформы к данным коммутаторам необходимо убедиться, что протокол PAgP для согласования не используется; иными словами, следует использовать режим `on` для форсирования использования EtherChannel-канала.

в) Выбор алгоритма балансировки нагрузки (*необязательно*)

Система COS	`set port channel all distribution {ip | mac |session} {source | destination | both}`
Система IOS	`port-channel load-balance` *method* (режим глобальной конфигурации)

Выбор метода (*method*) балансировки нагрузки на основе:

- IP-адреса отправителя — `ip source` (COS) или `src-ip` (IOS);
- IP-адреса получателя — `ip destination` (COS) или `dst-ip` (IOS);
- IP-адресов отправителя и получателя — `ip both` (COS) или `src-dst-ip` (IOS);
- MAC-адреса отправителя — `mac source` (COS) или `src-mac` (IOS);
- MAC-адреса получателя — `mac destination` (COS) или `dst-mac` (IOS);
- MAC-адресов отправителя и получателя — `mac both` (COS) или `src-dst-mad` (IOS);
- номера порта отправителя — `session source` (COS) или `src-port` (IOS);
- номера порта получателя — `session destination` (COS) или `dst-port` (IOS);
- номеров портов отправителя и получателя — `session both` (COS) или `src-dst-port` (IOS).

Совет

Балансировка нагрузки, основанная на указании параметров `session` или `port` (т.е. указании сеанса или номера порта), доступна только в блоке Supervisor 2 коммутатора Catalyst 6000 с *функциональной платой политики 2 (Policy Feature Card 2)* (как в системе COS, так и в IOS).

4. Использование протокола LACP на EtherChannel-каналах (*необязательно; только для COS-коммутаторов*).

а) Установка системного приоритета.

Система COS	`set lacp-channel system-priority` *value*
Система IOS	Нет

Системный идентификатор (ID) представляет собой восьмибайтовое число, сформированное из двухбайтового значения (*value*) приоритета (от 1 до 65535, стандартно 32768) с последующим указанием MAC-адреса. Наименьшее значение соответствует наивысшему приоритету. Коммутатор с наименьшим системным идентификатором способен в любое время управлять активным участием портов в EtherChannel.

б) Установка приоритета для отдельных портов.

Система COS	`set port lacp-channel mod/ports port-priority value`
Система IOS	Нет

Коммутатор с наименьшим значением системного идентификатора (ID) для использования в группе EtherChannel выбирает порты с наименьшими значениями идентификаторов. Идентификатор порта состоит из двухбайтового значения приоритета (от 1 до 255; стандартно 128) с последующим двухбайтовым номером порта. Порты, которые не могут использоваться, переводятся в состояние готовности (standby) и остаются неактивными до тех пор, пока другой канал в EtherChannel не выйдет из строя.

в) Группирование портов путем установки административных ключей.

Система COS	`set port lacp-channel mod/ports [admin-key]`
Система IOS	Нет

Порты, которые потенциально могут войти в состав EtherChannel, должны иметь одинаковые административные ключи (*admin-key*) (от 1 до 65535). Значение административного ключа может быть назначено не более чем восьми портам. Порты, имеющие уникальное значение, рассматриваются как неспособные и не входят в состав EtherChannel.

Стандартно каждая группа из четырех последовательных портов в модуле имеет одинаковое значение ключа. Значения ключа могут быть только локальными. Однако порты с одинаковыми значениями на одном коммутаторе потенциально способны формировать EtherChannel с портами, совместно использующими иное общее значение ключа на другом коммутаторе.

Если значение параметра *admin-key* не указано, то коммутатор выбирает неиспользуемое, уникальное значение для перечисленных портов. Если значение указано и уже используется, назначенные порты получают другое уникальное значение ключа.

г) Установка режима EtherChannel.

Система COS	`set port lacp-channel mod/ports mode {on | off | active passive}`
Система IOS	Нет

Протокол LACP может быть сконфигурирован в одном из режимов: **on** (включен, EtherChannel-канал используется, но LACP-пакеты не отправляются), **off** (выключен; EtherChannel-порт отключен), **desirable** (желательно; коммутатор готов к активному формированию EtherChannel; LACP-пакеты

отправляются) или auto (коммутатор готов к пассивному формированию EtherChannel, LACP-пакеты не отправляются; стандартный режим).

Совет

Несмотря на то что протоколы PAgP и LACP несовместимы и неспособны к взаимодействию, EtherChannel-канал может быть сформирован между коммутаторами, на модуле одного из которых используется протокол PAgP, а на модуле другого — LACP. В таком случае необходимо перевести PAgP- и LACP-коммутаторы в режим on. Для согласования EtherChannel-канала не будет использоваться ни один из данных протоколов, но канал будет сформирован.

Пример конфигурирования EtherChannel

На рис. 4.1 представлена диаграмма сети, иллюстрирующая данный пример. В коммутаторе имеются три линейные карты с Ethernet-портами. В модулях 4 и 5 для агрегирования портов используется протокол PAgP, а в модуле 6 — LACP. Для демонстрации того, что EtherChannel-канал может распределяться по нескольким линейным платам, EtherChannel-порт формируется из портов 4/1, 4/2, 5/1 и 5/2. Для динамического объединения портов в данном EtherChannel-порту используется протокол PAgP в режиме desirable. В режиме без пауз требуется наличие на данном конце канала PAgP спикера (PAgP speaker), прежде чем будет сформирован EtherChannel. Оба IP-адреса — как отправителя, так и получателя — распределяют трафик по объединенным портам.

Рис. 4.1. Диаграмма сети для примера конфигурации EtherChannel-канала

Второй EtherChannel-порт настраивается на использование протокола LACP. Приоритет системы в LACP устанавливается равным 8192, чтобы коммутатор получил наивысший приоритет для принятия решения. Порты 6/1, 6/2, 6/7 и 6/8 принадлежат административному ключу LACP 101, формируя общий агрегированный канал. Портам 6/1 и 6/2 назначен приоритет 100, который меньше стандартного 128. Данные порты используются в LACP-банке первыми. Если по каким-либо причинам порты 6/7 и 6/8 непригодны для использования в EtherChannel, они переходят в состояние готовности и используются при выходе из строя других портов. Каждый порт, входящий в банк, переведен в активный LACP-режим и готов согласовать EtherChannel с коммутатором на дальнем конце.

Система COS
```
set channelprotocol pagp 4
set channelprotocol pagp 5
set channelprotocol lacp 6
set port channel 4/1-2,5/1-2
set port channel 4/1 mode desirable non-silent
set port channel all distribution ip both
set lacp-channel system-priority 8192
set port lacp-channel 6/1-2 port-priority 100
set port lacp-channel 6/1-2,6/7-8 101
set port lacp-channel 6/1-2,6/7-8 mode active
```

Система IOS	`interface fastethernet 4/1`
	`channel-group 100 mode desirable non-silent`
	`interface fastethernet 4/2`
	`channel-group 100 mode desirable non-silent`
	`interface fastethernet 5/1`
	`channel-group 100 mode desirable non-silent`
	`interface fastethernet 5/2`
	`channel-group 100 mode desirable non-silent`
	`port-channel load-balance src-dst-ip`

Отображение информации о каналах EtherChannel

В табл. 4.3 перечислены некоторые команды, которые можно использовать для отображения полезной информации о каналах EtherChannel.

Таблица 4.3. Команды коммутатора для отображения сведений о каналах EtherChannel

Функция отображения	Операционная система коммутатора	Команда
Протокол EtherChannel, используемый в каждом модуле	COS	`show channelprotocol`
	IOS	Нет
Возможности формирования EtherChannel на каждом модуле	COS	`show port capabilities` *[mod [/port]]*
	IOS	Нет
ID-номера EtherChannel	COS	`show channel group` *admin-group* или `show lacp-channel group` *admin-key*
	IOS	Нет
Балансировка нагрузки EtherChannel	COS	`show channel` *[channel-id]* `info`
	IOS	`show etherchannel` *[channel-group]* `load-balance`
Использование трафика EtherChannel	COS	`show channel traffic` *[channel-id]* или `show lacp-channel traffic` *[channel-id]*
	IOS	`show pagp` *[group-number]* `counters`
Исходящий порт для адреса или номера порта	COS	`show channel hash` *channel-id [src | dest | src dest]* или `show lacp-channel hash` *channel-id [src | dest | src dest]*
	IOS	Нет

Совет

При отладке каналов EtherChannel, который не формируется по какой-либо причине, следует помнить, что все порты в группе должны иметь одинаковые атрибуты. Например, у всех портов должна быть одинаковая скорость, дуплексность, VLAN-сеть (или собственная VLAN-сеть для магистрального канала), магистральный режим и инкапсуляция, допустимый VLAN-диапазон и другие.

Команды, перечисленные в табл. 4.3, предоставляют большое количество сведений о сформированных EtherChannel-каналах. Чтобы убедиться, что все порты сконфигурированы согласованно, следует использовать другие команды show, которые отображают атрибуты портов. Кроме того, иногда приходится искать в конфигурации коммутатора настройки портов, которые не являются идентичными.

4.5: порты Token Ring

- Порты Token Ring способны работать на скорости 4 или 16 Мбит/с в полудуплексном или дуплексном режиме.
- Порты Token Ring способны функционировать в качестве Token Ring-концентраторов или конечных станций.
- Ниже приведены поддерживаемые мостовые режимы.
  - *Мостовая маршрутизация от отправителя (Source-Route Bridging — SRB)* — поля маршрутной информации (Routing Information Fields — RIF) переправляют пакеты от колец к мостам, вместо того чтобы изучать MAC-адреса. Поддерживаются обе версии протокола STP — IBM и IEEE.
  - *Прозрачная маршрутизация от отправителя (Source-Route Transparent — SRT)* — SRB-маршрутизация, объединенная с прозрачной мостовой передачей. Пакеты с RIF-полями передаются с помощью SRB, тогда как пакеты без RIF-полей передаются через прозрачные мосты. Поддерживается только IEEE-версия протокола STP.
  - *Коммутация маршрута от отправителя (Source-Route Switching — SRS)* — пакеты могут переключаться на основании MAC-адресов или RIF-полей

Конфигурация функции

1. Назначение порту описательного имени (*необязательно*).

Система COS	`set port name mod/port [port-name]`
Система IOS	Нет

Описание — `port-name` (текстовая строка) — назначается порту для удобства пользователя. Как правило, описание включает в себя ссылку на расположение, функцию или пользователя данного порта.

2. Установка скорости порта.

Система COS	`set port speed mod/port {4 | 16 | auto}`
Система IOS	Нет

Скорость порта можно установить равной 4 или 16 Мбит/с. Если используется ключевое слово `auto` (стандартное значение), порт автоматически определяет скорость подключенного кольца. При изменении скорости в порту он покидает кольцо и входит в него с новой скоростью.

Внимание!

Если стандартное автоопределение скорости не используется, следует внимательно отнестись к выбору корректной скорости для подсоединенного кольца. При отличающихся скоростях порта и кольца порт отправляет сигнальные фреймы (beacon frames), в результате чего кольцо теряет работоспособность.

3. Установка режима порта.

Система COS	`set tokenring portmode mod/port {auto | fdxcport | hdxcport | fdxstation | hdxstation | ziro}`
Система IOS	Нет

В зависимости от того, как используется порт, его режим может быть установлен в одно из следующих значений: `auto` (автоматическое определение режима соединения; стандартный режим), `fdxcport` (функционирует в качестве дуплексного концентратора при подключении к одной станции), `hdxcport` (функционирует в качестве полудуплексного концентратора при подключении к одной станции), `fdxstation` (функционирует в качестве дуплексной станции при подключении к другому Token-Ring-коммутатору), `hdxstation` (функционирует в качестве полудуплексной станции при подключении к блоку MSAU) или `ziro` (функционирует в качестве входа-выхода кольца при использовании волоконно-оптического соединения).

4. Использование *раннего освобождения маркера (Early Token Release — ETR)* *(необязательно)*.

Система COS	`set tokenring etr` *mod/port*`: {enable | disable}`
Система IOS	Нет

Функция ETR позволяет порту отправить свой маркер в кольцо немедленно после передачи (стандартное значение — `enable`). Обычно порт должен ожидать возвращения маркера до тех пор, пока передаваемый фрейм полностью не обойдет кольцо (`disable`). Раннее освобождение маркера возможно только на портах, входящих в 16-мегабитовые кольца.

5. Управление использованием фреймов-исследователей (explorer frames) *(необязательно)*.

   a) Сокращение количества *широковещательных всех маршрутов (All Routes Explorer — ARE) (необязательно)*.

Система COS	`set tokenring reduction {enable | disable}`
Система IOS	Нет

Если в сети Token Ring есть параллельные маршруты, то ARE-фреймы распространяются лавинно по всем возможным маршрутам. Количество этих фреймов можно сократить (`enable`; стандартно), выбирая идентичные анализаторы, которые уже были получены в кольце в точке подключения порта.

б) Регулировка количества входящих ARE-фреймов (*необязательно*).

Система COS	`set tokenring explorer-throttle` *mod/port* `maximum-explorers`
Система IOS	Нет

Количество ARE-фреймов ограничено параметром `maximum-explorers` (количество фреймов в секунду; стандартное значение равно нулю или регулировка отключена). По достижении порогового значения в течение интервала длительностью 1 секунда дополнительные ARE-фреймы, полученные на этом порту, отклоняются.

6. Реакция на возникновение ошибок на порту (*необязательно*).

а) Использование мониторинга незначительных ошибок (*необязательно*).

Система COS	`set station softerror` *mod/port* `{disable | enable}`
Система IOS	Нет

При стандартных настройках мониторинг незначительных ошибок отключен. При его включении на модуле или порту накапливаются статистические данные. Как правило, незначительные ошибки возникают в кольце, указывая на некритические отказы.

б) Установка порога незначительных ошибок (*необязательно*).

Система COS	`set station softerror` *mod/port* `threshold` *thres-num* `interval` *int-num*
Система IOS	Нет

Порог для количества незначительных ошибок от одной станции может устанавливаться на уровне `thres-num` (от 1 до 255, стандартно 100) ошибок в интервале времени, равном *int-num* (от 0 до 65534 секунд; стандартно 60 секунд). При достижении порогового значения и в случае, если интервал не равен нулю, коммутатор отправляет прерывание "soft error exceeded" (превышение порога ошибок).

в) Удаление из кольца станции, создающей ошибки (*необязательно*)

Система COS	`clear station` *mod/port mac-addr*
Система IOS	Нет

Если выясняется, что какая-либо станция производит большое количество незначительных ошибок и понижает производительность кольца, то она может стать кандидатом на удаление из этого кольца. MAC-адрес станции (*mac-addr*) указывается в шестнадцатеричном формате (00-11-22-33-44-55). Следует заметить, что коммутатор отправляет станции MAC-фрейм "Remove $station" (удалить станцию), который вынуждает ее отключиться от кольца.

г) Автоматическое отключение порта при потере конфигурации (*необязательно*)

Система COS	`set tokenring configlost` *mod/port* `[threshold` *thresh num*`] [interval` *int-num*`]`
Система IOS	Нет

Потеря конфигурации (configuration loss) происходит, когда порт входит в кольцо, передает данные, а затем по какой-либо причине закрывается или удаляется из кольца. Пороговое количество потерь конфигурации устанавливается на уровне параметра `threshold-num` (от 0 до 100, стандартно 5) потерь в течение интервала времени длительностью `int-num` (от 0 до 99 минут; стандартно 10 минут). По достижении порогового значения коммутатор отключает данный порт. Чтобы его снова использовать, порт необходимо повторно активизировать с помощью команды `set port enable`.

7. Использование фильтра для контроля MAC-адресов или протоколов в поступающих и порт фреймах (*необязательно*).

Система COS	`set port filter mod/port {mac addr | protocol-type} {permit deny}`
Система IOS	Нет

Можно разрешить (`permit`) или запретить (`deny`) поступление на какой-либо порт коммутатора фреймов, содержащих определенный MAC-адрес (`mac-addr`) отправителя или получателя, введенный в каноническом (`00-11-33-44-55`) или неканоническом (`00:11:22:33:40:55`) формате, или тип протокола (`protocol-type`).

Значение параметра `protocol-type` может быть четырехзначным шестнадцатеричным числом (например, 0x1234) для Ethernet-типа (см. приложение "Номера портов, протоколов и другие стандартные номера"), четырехзначным шестнадцатеричным числом для SNAP (например, 0x1234) или двузначным шестнадцатеричным числом для *точки доступа к службе получателя (Destination Service Access Point — DSAP)* (например, 0x12), всегда оканчивается на 00.

Для каждого Token Ring-порта может быть определено до 16 фильтров MAC-адресов и 16 фильтров протоколов (8 SAP- и 8 DSAP-фильтров).

Пример конфигурирования Token Ring-порта

Порт 2/5 Token Ring-коммутатора подключен к порту 3745 FEP в центре обработки данных на скорости 16 Мбит/с. Порт настроен на автоматическое определение своей роли в кольце. Включено ранее освобождение маркера. Кроме того, порт 2/5 настроен на отключение в случае возникновения пяти или более потерь конфигурации в течение пятнадцатиминутного интервала.

Порт 2/6 коммутатора подключен к *модулю многостанционного доступа (MultiStation Access Unit — MSAU)*, который также подключен к нескольким узлам. Этот порт для подключения к MSAU-устройству устанавливается в режим `bdstation`. В 16 Мбит/с-кольце используется ранее освобождение маркера. Для всего коммутатора включена функция сокращения количества ARE-фреймов.

Система COS	`set port name 2/5 3745 FEP`
	`set port speed 2/5 16`
	`set tokenring portmode 2/5 auto`
	`set tokenring arr 2/5 enable`
	`set tokenring configloss 2/5 threshold 5 interval 15`
	`set port name 2/6 1st floor MSAU`
	`set port speed 2/6 16`

	`set tokenring portmode 2/6 hdxstation`
	`set tokenring str 2/5 enable`
	`set tokenring reduction enable`

Система IOS	Нет

Отображение сведений о Token Ring-каналах

В табл. 4.4 перечислены команды коммутатора, которые можно использовать для отображения полезной информации о каналах Token Ring.

Таблица 4.4. Команды коммутатора для отображения сведений о каналах Token Ring

Функция отображения	Операционная система коммутатора	Команда
Состояние порта Token Ring	COS	`show tokenring`
	IOS	Нет
Состояние станций в кольце	COS	`show station controltable` *[mod[/port]]*
	IOS	Нет
Порядок подключения станций в кольце	COS	`show station orderstable` *[mod[/port]]*
	IOS	Нет
Настройка обработки незначительных ошибок	COS	`show station softerror config` *[mod[/port]]*
	IOS	Нет
Статистика незначительных ошибок	COS	`show station softerror counters mod/port` *[mac-addr]*
	IOS	Нет
Фильтры Token Ring	COS	`show port filter` *[mod[/port]]* `[canonical]`
	IOS	Нет

4.6: ATM LANE

- Технология LANE обеспечивает эмуляцию сети IEEE 802.3 Ethernet или IEEE 802.5 Token Ring в ATM-среде. Эта технология может применяться для транспортировки данных обычных LAN-сетей через ATM-магистраль или ATM WAN-среду.
- В технологии LANE для сегментации трафика в логические сети внутри ATM-домена используются *эмулированные локальные сети (emulated LAN)*, или ELAN-сети

- В состав сети LANE входит несколько логических компонентов, каждый из которых настраивается на маршрутизаторе, коммутаторе или ATM-коммутаторе:
  - *Конфигурационный сервер эмуляции локальной сети (LAN Emulation Configuration Server — LECS)* — центральная точка административного контроля для всех ELAN-сетей в домене. Сервер LECS поддерживает базу данных ELAN-сетей и ATM адресов LANE-серверов, управляющих каждой ELAN-сетью. (В каждом административном домене имеется только один LECS-сервер.)
  - *Сервер эмуляции локальной сети (LAN Emulation Server — LES)* — центральная контрольная точка для всех LANE-клиентов в ELAN-сети. LES-сервер обеспечивает для каждого LANE-клиента преобразование MAC-адресов в ATM-адреса *точек доступа к сетевой службе (Network Service Access Point — NSAP)*. В каждой ELAN-сети имеется только один LES-сервер.
  - *Сервер широковещательных и неизвестных сообщений (Broadcast and Unknown Server — BUS)* обрабатывает все широковещательные пакеты, отправленные с LANE-узла. LANE-клиент должен перенаправлять серверу BUS все широковещательные или многоадресатные пакеты от конечных пользователей. BUS-сервер затем может распространить полученные широковещательные пакеты всем остальным LANE-клиентам в домене.
  - *Клиент LANE-эмуляции (LAN Emulation Client — LEC)* обеспечивает основную ELAN-функцию на границе ATM-сети. Клиент LEC эмулирует интерфейс к обычной локальной сети и обеспечивает передачу данных, преобразование адресов и регистрацию MAC-адресов в других LANE-компонентах. Клиент LEC необходим везде, где используются адреса сетевого уровня.
- Для обеспечения резервирования можно построить в сети несколько LANE-компонентов. *Простой протокол резервирования серверов (Simple Server Redundancy Protocol — SSRP)* реализует процесс обмена информацией между активными и резервными компонентами, поэтому в сети может быть изолирована точка сбоя.

Внимание!

В ATM-адресах используется NSAP-формат, двадцатибайтовая значения. Как правило, NSAP-адреса записываются в виде групп, состоящих из четырех шестнадцатеричных значений, разделенных точками. Левая и правая группы адреса обычно состоят из двух шестнадцатеричных чисел. Блоки, входящие в состав адреса, описаны ниже.

Префикс (prefix) — тринадцатибайтовое поле, уникально идентифицирующее каждый ATM-коммутатор в сети. В ATM-коммутаторах Cisco используется предопределенное семибайтовое значение 47.0091.8100.0000, за которым следует шестибайтовый MAC-адрес коммутатора.

Идентификатор конечной системы (End-System Identifier — ESI) — шестибайтовое поле, уникально идентифицирующее каждое устройство, подсоединенное к ATM-коммутатору. Как правило, идентификатором является шестибайтовый MAC-адрес устройства (например, интерфейса ATM-маршрутизатора или LANE-модуля).

Селектор (selector) — однобайтовое поле, идентифицирующее процесс, выполняющийся в ATM-устройстве. В качестве значения селекторов в устройствах Cisco обычно применяется номер подынтерфейса ATM. Прежде чем конфигурировать байт селектора на коммутаторе Cisco, необходимо преобразовать номер подынтерфейса ATM из десятичной формы в шестнадцатеричную.

Значение префикса всегда предоставляется ATM-коммутатором. ESI-значение определяется из MAC-адреса ATM-интерфейса. LEC (MAC-адрес), LES (MAC-адрес + 1), BUS

(MAC-адрес + 2) и LECS (MAC-адрес + 3). Селектор представляет собой номер подынтерфейса ATM, за исключением селектора LECS-сервера, который всегда должен конфигурироваться на главном ATM-интерфейсе (селектор 00)

Конфигурация функции

1. Доступ к сеансу ATM LANE-модуля

Система COS	`session` *module*
Система IOS	Нет

Запускается Telnet-сеанс с LANE-модулем в блейде с номером *module*. Для определения номера блейда используется команда **show module**. После того как откроется сеанс, происходит обмен данными с CLI-интерфейсом операционной системы IOS LANE-модуля. Для возврата к командную строку COS-коммутатора необходимо ввести команду **exit**.

2. Указание управляющих *постоянных виртуальных каналов* (*Permanent Virtual Circuits — PVC*).

   а) Выбор главного ATM-интерфейса.

Система COS	Нет
Система IOS	**interface atm** 0 (режим глобальной конфигурации)

   б) Выбор предпочтительного PHY-интерфейса (*only for uplinks*)

Система COS	Нет
Система IOS	**atm preferred phy** {A | B} (режим конфигурирования интерфейса)

В ATM-модуле со сдвоенным интерфейсом в качестве предпочтительного можно выбрать либо физический интерфейс A либо B.

   в) Указание сигнального ATM PVC-канала

Система COS	Нет
Система IOS	**atm pvc** *vcd* 0 5 **qsaal** (режим конфигурирования интерфейса)

В сигнальном PVC-канале используется сигнализация QSAAL и, как правило, VPI/VCI-значения 0/5. *vcd* — дескриптор виртуального канала, произвольное число (от 1 до 2047), которое применяется для уникальной идентификации PVC-канала.

   г) Указание интерфейса **ILMI** PVC-канала.

Система COS	Нет
Система IOS	**atm pvc** *vcd* 0 16 **ilmi** (режим конфигурирования интерфейса)

Промежуточный интерфейс локального управления[1] (*Interim Local Management Interface — ILMI*) — протокол, используемый между LANE-модулем и ATM-коммутатором для обмена данными и установки различных ATM-параметров. В частности, ILMI-интерфейс может использоваться для передачи адреса LECS-сервера какому-либо коммутатору в LANE-среде. ILMI-интерфейс обычно конфигурируется на канале VPI/VCI 0/16. Для идентификации PVC-канала значение поля *vcd* может быть произвольным.

3. Отображение стандартных LANE-адресов на ATM PHY-интерфейсе.

Система COS	Нет
Система IOS	**show lane default-atm-addresses**

С помощью этой команды отображаются стандартные NSAP-адреса для LANE-компонентов. Если LANE-модуль получил префиксную часть адреса от ATM-коммутатора, то отображается весь NSAP-адрес. В противном случае отображается только ESI- или MAC-часть адреса. Данные NSAP-адреса используются в дальнейших этапах конфигурирования LANE. Ниже приводится пример информации, полученной от LANE-модуля.

```
show lane default-atm-addresses
interface ATM0:
LANE Client:          47.0091810000000061705A01.00503EDD0C10.**
LANE Server:          47.0091810000000061705A01.00503EDD0C11.**
LANE Bus:             47.0091810000000061705A01.00503EDD0C12.**
LANE Config Server:   47.0091810000000061705A01.00503EDD0C13.00
note: ** is the subinterface number byte in hex
```

4. Указание LECS-сервера (*необязательно*).

а) Имя базы данных LECS-сервера.

Система COS	Нет
Система IOS	**lane database** *database-name* (режим глобальной конфигурации)

База данных LECS-сервера получает имя в параметре *database-name* (строка от 1 до 32 символов).

б) Указание ELAN сети и ее LES-сервера.

Система COS	Нет
Система IOS	**lane** *elan-name* **server-atm-address** *atm-address* **[restricted] [index** *index***]**

ELAN-сеть с именем, указанным в параметре *elan-name* (строка от 1 до 32 символов), принадлежит к LES-серверу с адресом *atm-address* (двадцатибайтовый NSAP-адрес). NSAP-адрес может быть получен при помощи команды **show lane default-atm-addresses** на маршрутизаторе или LANE-модуле, на котором будет располагаться LES-компонент. Для того чтобы ограничить состав ELAN-сети

[1] *Этот интерфейс также часто называют* временным. — *Прим. ред.*

только теми клиентами, которые явно указаны в базе данных, можно использовать ключевое слово `restricted`.

Внимание!

Чтобы использовать протокол SSRP для поддержки резервирования, можно определить для ELAN-сети несколько LES-компонентов. Для того чтобы назначить каждому LES-серверу приоритет (0 — наивысший приоритет), необходимо использовать ключевое слово `index`.

- Указание имени стандартной ELAN-сети. *(Для неограниченного состава ELAN-сети).*

Система COS	Нет
Система IOS	`default-name` *elan-name*

В ELAN-сети с неограниченным составом (unrestricted membership) любой клиент, пытающийся зарегистрироваться на LECS-сервере, будет подключен к ELAN-сети, имя которой указано в параметре `elan-name` (строка от 1 до 32 символов).

- Определение необходимых LEC-клиентов и их ELAN-сети. *(Ограниченный состав ELAN-сети.)*

Система COS	Нет
Система IOS	`client-atm-address` *atm-address-template* `name` *elan-name*

В ELAN-сети с ограниченным составом (Restricted ELAN membership) любой клиент, пытающийся зарегистрироваться на LECS-сервере, должен быть определенным образом идентифицирован по своему ATM-адресу (*atm-address* — двадцатибайтовое значение, 40 шестнадцатеричных цифр; звездочка (*) может соответствовать любой цифре, многоточие (...) может соответствовать любому количеству цифр). Соответствующий клиент подключается к ELAN-сети с именем, определенным в параметре `elan-name` (строка от 1 до 32 символов).

в) Включение LECS-сервера. Необходимая конфигурация включает в себя несколько этапов.

- Выбор главного ATM-интерфейса.

Система COS	Нет
Система IOS	`interface atm 0` (режим глобальной конфигурации)

- Использование определенной LECS-базы данных.

Система COS	Нет
Система IOS	`lane config database` *database-name* (режим конфигурирования интерфейса)

- Указание ATM-адреса LECS-сервера.

Внимание!

Чтобы использовать резервные LECS-компоненты, необходимо в ATM-коммутаторе задать ATM-адрес каждого LECS-сервера. После того как каждый LEC-клиент инициализируется, он через UNI-интерфейс получает от ATM-коммутатора список всех адресов LECS-серверов. Если внутри ELAN-домена используется множество ATM-коммутаторов, все они должны иметь идентичный список LECS-адресов, расположенных в одинаковом порядке. В дополнение к этому все избыточные LECS-серверы должны иметь идентичную базу данных.

Для LECS-сервера имеется выбор из трех различных ATM-адресов.

- Чтобы использовать автоматический или предопределенный адрес, применяются указанные ниже команды.

Система COS	Нет
Система IOS	`lane config auto-config-atm-address` (режим конфигурирования интерфейса)

Адрес LECS-сервера формируется из значения, предоставленного командой `show lane default-atm-addresses`.

- Для того чтобы использовать фиксированный LECS-адрес, применяется приведенная ниже команда.

Система COS	Нет
Система IOS	`lane config fixed config-atm-address` (режим конфигурирования интерфейса)

LECS-адрес — это фиксированный NSAP-адрес
47.0079000000000000000000.00A03E000001.00

- Для использования специфического ATM-адреса применяется приведенная ниже команда.

Система COS	Нет
Система IOS	`lane config config-atm-address` *nsap-address* (режим конфигурирования интерфейса)

LECS-сервер получает адрес, указанный в параметре *nsap-address* (шестнадцатибайтовое значение или 40 шестнадцатеричных цифр). Адрес может быть задан в качестве шаблона с использованием символа звездочки (*), соответствующего какой-либо шестнадцатеричной цифре, или многоточия (...), которое соответствует любому количеству начальных, средних или конечных шестнадцатеричных цифр.

5. Указание LES-BUS-пары *(необязательно)*.

a) Выбор подинтерфейса ATM.

Система COS	Нет
Система IOS	`interface atm 0.`*subinterface* (режим глобальной конфигурации)

Для LES/BUS-компонентов можно использовать произвольный номер подинтерфейса (*subinterface*). Вместе с тем существует возможность настроить на подинтерфейсе только один LES/BUS-сервер, обслуживающий только одну ELAN-сеть. В дополнение к этому для поддержки ELAN-сети необходима только одна пара LES-BUS.

б) Включение LES/BUS-сервера.

Система COS	Нет
Система IOS	`lane server-bus ethernet` *elan-name* (режим конфигурирования интерфейса)

Серверы LES и BUS создаются для ELAN-сети с именем, определенным в параметре *elan-name* (строка от 1 до 32 символов). Сеть с LAN-эмуляцией функционирует как Ethernet-сеть.

6. Определение LEC-клиента (*необязательно*).

а) Выбор ATM-подинтерфейса.

Система COS	Нет
Система IOS	`interface atm 0.`*subinterface* (режим глобальной конфигурации)

Для LEC-компонента можно использовать произвольный номер подинтерфейса (*subinterface*). LEC-клиент может настраиваться на тот же подинтерфейс, что и LES/BUS-пара, хотя такая конфигурация не является обязательной. Однако на одном подинтерфейсе можно создать только один LEC-клиент, который обслуживает одну ELAN-сеть.

б) Включение клиента LEC.

Система COS	Нет
Система IOS	`lane client ethernet` *vlan-id [elan name]* (режим конфигурирования интерфейса)

Клиент LEC конфигурируется для эмуляции Ethernet-сети. Когда он подключается к ELAN-сети, в базе данных LECS сервера уже имеется имя ELAN-сети для данного клиента. Имя ELAN-сети — *elan-name* — может быть задано так, что LEC представит имя ELAN-сети LECS-серверу для дополнительного сравнения. VLAN-сеть с номером *vlan-id* будет связана с данной ELAN-сетью мостом.

Пример конфигурации технологии LANE для ATM-среды

LANE-модуль коммутатора Catalyst конфигурируется для использования в LANE-среде бочонка. После того как сконфигурированы сигнальный и ILMI PVC-каналы, LANE-модуль готов дополнить все части стандартных ATM-адресов. Стандартные адреса локального LANE-модуля показаны в примере, хотя стандартные адреса других устройств также собраны. База данных сервера LECS конфигурируется как `bootflash:lecs.db`. ELAN-сеть с

именем radiology (радиология) конфигурируется с ограниченным составом. В целях избыточности задается два адреса LES-серверов: LES-сервер локального LANE-модуля с приоритетом ноль и LES-сервер другого устройства с приоритетом, равным единице. Для простоты указывается только один LEC-клиент, способный подключиться к ELAN-сети отделения радиологии — LEC на локальном LANE-модуле.

Вторая сеть с именем surgery (хирургия) также конфигурируется с неограниченным составом. Стандартно LEC-клиент подключается к ELAN-сети хирургического отделения.

LECS-сервер конфигурируется на интерфейсе atm 0, LES/BUS-пара для отделения радиологии — на интерфейсе atm 0.1, а LES/BUS-пара для хирургического отделения — на интерфейсе atm 0.2. Наконец, LEC-клиент в ELAN-сети радиологического отделения настраивается на интерфейсе atm 0.3 для маппингового соединения с сетью VLAN 10. Следует отметить, что ATM-адреса, перечисленные в базе данных, обладают корректным значением LANE-компонента в последней цифре NSAP-идентификатора (0 = LEC, 1 = LES, 2 = BUS, 3 = LECS). Значение селектора также имеет номер соответствующего подынтерфейса, используемого в конфигурации интерфейса.

Система COS	Нет
Система IOS	В режиме глобальной конфигурации необходимо ввести: **interface atm 0** далее в режиме конфигурирования интерфейса необходимо ввести: **atm pvc 1 0 5 qsaal** **atm pvc 2 0 16 ilmi** **no shutdown** **exit** **exit** **show lane default-atm-addresses** LANE Client:47.0091810000000E01E35A901.00E0FB1400C0.** LANE Server:47.0091810000000E01E35A901.00E0FB1400C1.** LANE Bus:47.0091810000000E01E35A901.00E0FB1400C2.** LANE Config Server.47.0091810000000E01E35A901.00E0FB1400C2.00 (Указанная выше команда также запускается на других LES- и LEC-маршрутизаторах и коммутаторах, поэтому в конфигурацию могут добавляться адреса LANE-компонентов.) **config terminal** **lane database hospital-db** **name radiology server-atm-address** **47.0091810000000E01E35A901.00E0FB1400C1.01 restricted** **index 0** **name radiology server-atm-address** **47.0091810000000E01E35A901.00E0FB121042.01 restricted** **index 1** **client-atm-address** **47.0091810000000E01E35A901.00E0FB1400C0.03 name** **radiology** **name surgery server-atm-address** **47.0091810000000E01E35A901.00E0FB1400C1.02 index 0** **name surgery server-atm-address** **47.0091810000000E01E35A901.00E0FB121041.02 index 0**

```
default-name surgery
exit

interface atm 0
description LECS for hospital ATM network
lane config database hospital-db
lane config auto-config-atm-address

interface atm 0.1
description LES/BUS for radiology ELAN
lane server-bus ethernet radiology

interface atm 0.2
description LES/BUS for surgery ELAN
lane server-bus ethernet surgery

interface atm 0.3
description LEC for radiology ELAN
lane client ethernet 10 radiology
```

Отображение информации об ATM LANE-компонентах

В табл. 4.5 перечислены некоторые команды, которые можно использовать для отображения полезной информации о LANE-компонентах ATM-среды.

Таблица 4.5. Команды коммутатора для отображения сведений об ATM LANE-компонентах

Функция отображения	Операционная система коммутатора	Команда
Состояние LECS-сервера	COS	Нет
	IOS	`show lane config`
Отображение базы данных LECS	COS	Нет
	IOS	`show lane database`
Состояние устройства LES	COS	Нет
	IOS	`show lane server`
Состояние сервера BUS	COS	Нет
	IOS	`show lane bus`
Состояние устройства LEC	COS	Нет
	IOS	`show lane client`
Стандартные NSAP-адреса сети ATM	COS	Нет
	IOS	`show lane default-atm-addresses`

Совет

Если при передаче трафика через устройство LEC возникают проблемы, рекомендуется использовать команду show lane client для проверки правильности подключений клиента LEC к ELAN-сети. Информация, которая отображается при успешном подключении LEC клиента, приведена ниже.

```
lane-router# show lane client
LE Client ATM0.1 ELAN name: elan43 Admin: up State: operational
Client ID: 1            LEC up for 3 days 19 hours 6 minutes 40 seconds
Join Attempt: 9
HW Address: 0000.1499.6410 Type: ethernet Max Frame Size: 1516
        VLAN ID: 67
ATM Address: 47.009181000000000E01E35A901.00E014996410.01
VCD rxFrames txFrames Type        ATM Address
  0        0        0 configure
47.009181000000000E01E35A901.00E07B1400C3.00
 16        1   211037 direct
47.009181000000000E01E35A901.00E014996411.01
 87    40143        0 distribute
47.009181000000000E01E35A901.00E014996411.01
 90        0   819712 send
47.009181000000000E01E35A901.00E014996412.01
 91 999999999        0 forward
47.009181000000000E01E35A901.00E014996412.01
199 12033702 11492102 data
47.009181000000000E01E35A901.00E07B1400C2.01
```

Дополнительная литература

Рекомендуемые ниже источники представляют более подробную информацию по темам, рассматриваемым в этой главе.

Технология Ethernet

Web-сайт Чарльза Спуржина (Charles Spurgeon) об технологии Ethernet. www.host.ots.utexas.edu/ethernet/.

Charles Spurgeon. *Ethernet: The Definitive Guide*, O'Reilly and Associates

Технология Fast Ethernet

Решения корпорации Cisco с использованием портов 100 Мбит/с Fast Ethernet (Cisco Fast Ethernet 100 Mbps Solutions). www.cisco.com/warp/public/cc/so/neso/lnso/lemnac/fethl_so.htm

Технология Gigabit Ethernet

Сообщество Gigabit Ethernet: www.gigabit-ethernet.org.

Сообщество 10-Gigabit Ethernet: www.10gea.org/index.htm.

Стандарт IEEE 802.3ae: http://grouper.ieee.org/groups/802/3/ae/index.html.

Технология EtherChannel

Вопросы конфигурирования каналов FastEtherChannel на коммутирующих и маршрутизирующих устройствах Cisco (Understanding and Configuring FastEtherChannel on Cisco Switching and Routing Devices): www.cisco.com/warp/public/473/4.html.

Вопросы проектирования сетей с каналами FastEtherChannel (Understanding and Designing Networks with FastEtherChannel): www.cisco.com/warp/public/cc/techno/media/lan/ether/channel/prodlit/faste_an.htm.

LACP — рабочая группа по проблемам агрегирования каналов (LACP — Link Aggregation Task Force): http://grouper.ieee.org/groups/802/3/ad/.

Технология Token Ring

Стандарт Token Ring/IEEE802.5 (Token Ring/IEEE802.5): www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/tokenrng.htm.

Технология LANE сети ATM

Сети ATM корпорации Cisco®. Галина Дикер Пилщик. ИД "Вильямс", 2004.

Web-сайт Форума ATM (The ATM Forum): www.atmforum.com.

В этой главе...

- **5.1: коммутация третьего уровня.** В этом разделе описывается процесс коммутации третьего уровня и коммутирующие элементы, необходимые для ее реализации.
- **5.2: Ethernet-интерфейсы третьего уровня.** В разделе описываются этапы конфигурирования Ethernet-интерфейсов для обработки информации третьего уровня.
- **5.3: EtherChannel-каналы третьего уровня.** В настоящем разделе описан метод конфигурирования множества интерфейсов в виде одного логического канала, который можно настраивать для обработки информации третьего уровня.
- **5.4: WAN-интерфейсы.** В разделе описывается конфигурирование WAN-интерфейсов третьего уровня, установленных в коммутаторах Catalyst 5500 и 6500.
- **5.5: виртуальные интерфейсы.** В этом разделе описана методика конфигурирования логической VLAN-сети или BVI-интерфейса с целью реализации механизма обработки информации третьего уровня для членов VLAN-сети или мостовой группы.
- **5.6: таблицы маршрутизации.** В разделе представлено объяснение основного процесса пополнения и просмотра содержимого таблиц маршрутизации третьего уровня.

Глава 5

Конфигурирование интерфейсов третьего уровня

5.1: коммутация третьего уровня

- Коммутация третьего уровня представляет собой перемещение данных между устройствами с помощью таблиц или маршрутов, содержащих сетевые адреса третьего уровня.
- Чтобы осуществлять коммутацию третьего уровня, устройству требуется процессор коммутации третьего уровня, который может быть выполнен в виде отдельного модуля или платы.
- Для конфигурирования коммутирующих компонентов третьего уровня в процессоре коммутации третьего уровня используется операционная система IOS с поддержкой функций третьего уровня.
- Чтобы обеспечить возможность коммутации третьего уровня, в коммутаторе необходимо включить функцию маршрутизации для определенного протокола.
- Для обеспечения связи между различными сетями коммутатору требуются данные о доступных маршрутах к этим сетям.

Конфигурация функции

Для того чтобы выполнять на каком-либо устройстве коммутацию третьего уровня, необходимо наличие процессора коммутации, принимающего решения на основании адресации протоколов третьего уровня. Такое устройство работает под управлением операционной системы Cisco IOS, и конфигурирование осуществляется только в режиме конфигурации IOS. Процессор коммутации может быть выполнен в виде *модуля коммутации маршрутов (Route Switch Module — RSM)*, *функциональной платы коммутации маршрутов (Route Switch Feature Card — RSFC)*, *модуля многоуровневой коммутации (Multilayer Switch Module — MSM)*, *функциональной платы многоуровневой коммутации (Multilayer Switch Feature Card — MSFC)*, служебного модуля третьего уровня или блока, интегрированного в аппаратное обеспечение данного коммутатора, например, 3550 или 2948G-L3.

1. Доступ к процессору коммутации третьего уровня.

   Если процессор коммутации выполнен в виде платы, работающей в качестве подсистемы в COS-устройстве, то для конфигурирования необходимо получить доступ к этому устройству. Доступ к устройству осуществляется с помощью команды `session`. Она не требуется для коммутатора, работающего с операционной системой Supervisor IOS, поскольку с процессором третьего уровня устанавливается непосредственная связь. Ниже представлена команда доступа к процессору коммутации COS-коммутатора.

Система COS	`session` *mod* (в режиме привилегированного пользователя)

   Параметр *mod* определяет номер модуля процессора коммутации. Если местоположение процессора коммутации неизвестно, для его определения используется команда `show modules`.

Внимание!

Для плат MSFC и RSFC на блоке Supervisor в гнезде 1 номер модуля всегда равен 15. Для тех же плат на блоке Supervisor в гнезде 2 номер модуля всегда равен 16.

2. Включение процесса маршрутизации.

3. Для осуществления процесса коммутации третьего уровня одного наличия процессора недостаточно, процессор необходимо настроить. Настройка производится в режиме глобальной конфигурации путем включения процесса маршрутизации для необходимого протокола.

Система IOS	*protocol* `routing` (режим глобальной конфигурации)

   Эта команда активизирует процесс маршрутизации. Параметр *protocol* определяет протокол, который необходимо включить, например, `appletalk`, `ip` или `ipx`. При стандартных настройках маршрутизация отключена для всех протоколов, кроме IP, поэтому для коммутации IP-пакетов устройством третьего уровня эта команда необязательна.

   Если коммутатор настроен на выполнение *функции интегрированной маршрутизации и установления мостового соединения (Integrated Routing And Bridging — IRB)*, необходимо также включить процесс маршрутизации для любой мостовой группы, которой требуется доступ за пределы широковещательного домена. Чтобы включить коммутацию третьего уровня для мостовых групп, используется команда, приведенная ниже.

Система IOS	`bridge` *groupnumber* `route` *protocol* (режим глобальной конфигурации)

   Эта команда позволяет всем членам мостовой группы, указанной с помощью параметра *groupnumber*, использовать виртуальный интерфейс для обмена данными посредством коммутации третьего уровня. Параметр *protocol* определяет, какой протокол будет обрабатываться на третьем уровне.

Внимание!

Если указанная команда задана для конкретного протокола в мостовой группе, но адрес протокола виртуального интерфейса не назначен, то мостовые функции для этого протокола будут отключены

5.2: Ethernet-интерфейсы третьего уровня

- Коммутация третьего уровня требует наличия в коммутаторе интерфейса, который способен перенаправлять пакеты на основании адресации третьего уровня.
- Каждый интерфейс третьего уровня определяет отдельный широковещательный домен и, следовательно, отдельную сеть.
- После того как интерфейс третьего уровня настроен на какой-либо протокол, он может функционировать в качестве шлюза для других устройств в том же широковещательном домене.
- На некоторых коммутаторах можно настроить Ethernet-порт (интерфейс) в качестве интерфейса третьего уровня.

Конфигурация функции

Интерфейс третьего уровня представляет собой прямой маршрутизируемый интерфейс, который предназначен для обработки на третьем уровне пакетов, поступающих на этот интерфейс и покидающих его. Не все физические интерфейсы коммутаторов предназначены для работы в качестве интерфейса третьего уровня. На некоторых коммутаторах каждый порт является или может быть настроен для работы в качестве прямого маршрутизируемого порта. Ниже описаны этапы настройки таких интерфейсов для обработки данных третьего уровня.

1. Выбор физического интерфейса третьего уровня.

Система IOS	`interface type mod/port` (режим глобальной конфигурации)

Необходимо войти в режим глобальной конфигурации и воспользоваться этой командой для указания интерфейса и перехода в режим конфигурации интерфейса данного устройства. Необходимо указать тип интерфейса `ethernet`, `fastethernet`, `gigabitethernet` или `tengigabitethernet`. В параметре *mod/port* указывается модуль и номер порта интерфейса. Коммутаторы с фиксированной конфигурацией (fixed-configuration switches), такие, как 2948G L3, не имеют параметра модуля, а в коммутаторах 3550 номер модуля или всегда равен 0.

2. Конфигурирование интерфейса для операций третьего уровня.

Система IOS	`no switchport` (режим конфигурирования интерфейса)

Для многоуровневых IOS-коммутаторов, таких, как 4000 и 6000, использующих систему Supervisor IOS, или коммутатора 3550 порты можно настраивать для работы в качестве портов второго (порты коммутации) или третьего (маршрутизируемых)

уровня. Для настройки порта в качестве порта третьего уровня используется команда **no switchport**, которая отключает функции второго уровня и включает операции третьего.

Внимание!

При вводе команды **switchport** или **no switchport** порт отключается, а затем включается снова.

Внимание!

Стандартным режимом для порта в коммутаторах серий 4000 и 6000, использующих операционную систему Supervisor IOS, является маршрутизируемый. Порты коммутатор 3550 стандартно настроены на режим коммутации (switchport mode)

3. Конфигурирование протокольной информации.

а) Назначение IP-адреса.

Система IOS	**ip address** *address netmask* (режим конфигурирования интерфейса)

Когда интерфейс начинает функционировать как интерфейс третьего уровня, необходимо указать ему информацию о сети, подключенной к широковещательному домену. Для IP-сетей это означает, что интерфейсу нужно назначить IP адрес. Этот адрес станет адресом шлюза, используемым клиентами в широковещательном домене, к которому подключен интерфейс.

Внимание!

Несмотря на то что все интерфейсы коммутации-маршрутизации третьего уровня поддерживают протокол IP, не все они поддерживают другие протоколы. Например, коммутаторы 3550 и 4000 (система Supervisor IOS) поддерживают только конфигурацию протокола IP на физическом интерфейсе. Изучите рекомендации к используемым устройствам для получения более подробной информации.

б) Назначение номера сети IPX.

Система IOS	**ipx network** *netwerknumber* (режим конфигурирования интерфейса)

Для активизации на интерфейсе IPX-процесса необходимо в параметре *network-number* указать номер сети. Чтобы получить соответствующий номер сети и типа инкапсуляции, обратитесь к администратору сети NetWare.

в) Назначение кабельного диапазона и зоны AppleTalk.

Система IOS	**appletalk cable-range** *beginrange-endrange* **appletalk zone** *zonename* (режим конфигурирования интерфейса)

Для включения протокола AppleTalk необходимо указать кабельный диапазон, используя параметр *beginrange-endrange*, определяющий числа диапазона, а также с помощью команды **appletalk zone** настроить хотя бы одну зону.

Внимание!

Информация, представленная здесь для конфигурирования параметров протокола на интерфейсе третьего уровня, является минимально необходимой. Более подробные сведения, касающиеся конфигурации протоколов, приведены в David Hucaby and Steve McQuerry, *Cisco Field Manual: Router Configuration*, Cisco Press.

4. Включение интерфейса.

Система IOS	`no shutdown` (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является **shutdown** (отключен). Для того чтобы интерфейс функционировал, необходимо включить его с помощью команды `no shutdown`.

Проверка конфигурации

Для проверки настроенной конфигурации протокола на интерфейсе используется приведенная ниже команда.

Система IOS	`show` *protocol* `interface` *type mod/port* (режим привилегированного пользователя)

Параметром протокола являются **ip**, **ipx**, **appletalk** или другой сконфигурированный интерфейс.

В следующем примере демонстрируется конфигурация интерфейса Gigabit Ethernet 1/1 для осуществления обработки третьего уровня на коммутаторе Distribution_Switch_A (коммутатор уровня распределения). Этот интерфейс функционирует в качестве шлюза для всех клиентов, подключенных к коммутатору Access_Switch_A (коммутатор уровня доступа к сети). На рис. 5.1 представлена топология сети для рассматриваемого примера.

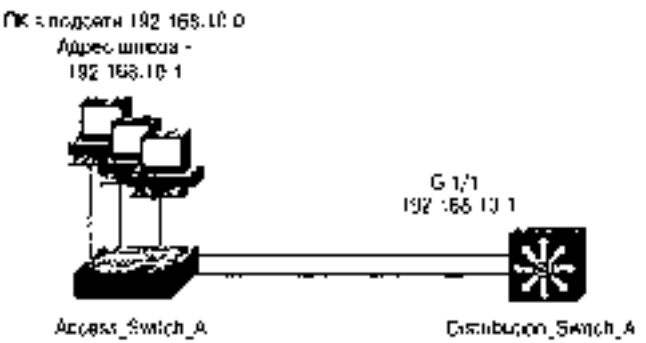


Рис. 5.1. Топология сети для примера по конфигурации интерфейса третьего уровня

Ниже приводится пример конфигурации коммутатора Distribution_Switch_A.

```
Distribution_Switch_A(config)#interface gigabitethernet 1/1
Distribution_Switch_A (config-if)#no switchport
Distribution_Switch_A (config-if)#ip address 192.168.10.1
255.255.255.0
```

```
Distribution_Switch_A (config-if)#no shut
Distribution_Switch_A (config-if)#end
Distribution_Switch_A #copy running-config startup-config
```

5.3: EtherChannel-каналы третьего уровня

- *Канал EtherChannel* представляет собой суммирование множества физических каналов в один логический.
- Одно логическое соединение называется *портом-каналом (port channel)*.
- На некоторых коммутаторах можно настроить такой канал для функционирования в качестве интерфейса третьего уровня.
- При назначении IP-адреса каналу он становится логическим интерфейсом третьего уровня.
- Если какой-либо канал выходит из строя, интерфейс канала остается доступным через другие каналы.
- Функции EtherChannel-канала третьего уровня подобны функциям группы EtherChannel второго уровня по распределению трафика и установке каналов.

Конфигурация функции

Канал EtherChannel предоставляет возможность связывать множество физических соединений в целях обеспечения большей пропускной способности для каналов, по которым транспортируется трафик нескольких узлов. Поскольку канал EtherChannel функционирует "почти" на физическом уровне, в один канал могут быть связаны несколько интерфейсов третьего уровня. После того как канал был сформирован, виртуальный интерфейс, который называется *каналом*, начинает функционировать как канал третьего уровня для всех его членов. Настройка группы EtherChannel включает в себя следующие этапы.

1. Доступ к процессору коммутации третьего уровня.

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к этому устройству. Для этого используется команда **session**. При использовании коммутатора, работающего с системой Supervisor IOS, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутации третьего уровня.

Система COS	**session** *mod* (режим привилегированного пользователя)

   Параметр *mod* указывает номер модуля процессора коммутации. Чтобы определить местоположение процессора в коммутаторе, если оно неизвестно, используется команда **show module**.

Внимание!

Доступ к процессору коммутации третьего уровня имеется только тогда, когда создается канал к объединительной плате на модуле служб третьего уровня. Все остальные интерфейсы третьего уровня настраиваются непосредственно с помощью программного обеспечения операционной системы IOS

2. Создание логического канала

Система IOS	`interface port-channel` *number* (режим глобальной конфигурации)

Для создания логического интерфейса канала эта команда вводится в режиме глобальной конфигурации. Интерфейс функционирует как интерфейс третьего уровня для всех членов канала. Параметр *number* определяет номер группы канала, который будет назначен каждому члену канала.

3. Настройка протоколов в канале

Система IOS	`ip address` *address netmask* (режим конфигурирования интерфейса)

Для настройки сетевой адресации на интерфейсе третьего уровня используется соответствующие команды. В приведенном выше примере демонстрируется конфигурирование IP адреса. Параметры для других протоколов приводится в разделе "5.2: Ethernet-интерфейсы третьего уровня"

Внимание!

При создании канала, который будет использовать адрес, в настоящее время настроенный на данном интерфейсе, прежде всего перед назначением адреса интерфейсу канала порта необходимо удалить этот адрес. О том, как удалить адрес протокола, рассказывается ниже

4. Назначение физических интерфейсов третьего уровня группе каналов

а) Выбор интерфейса

Система IOS	`interface type` *mod/port* (режим глобальной конфигурации)

Нужно выбрать интерфейс третьего уровня для назначения группе каналов. Поскольку создается канал третьего уровня, должен использоваться интерфейс также третьего уровня. Для коммутаторов, допускающих функционирование интерфейса в качестве интерфейса второго или третьего уровня, вводится команда по `switchport`, обеспечивающая функционирование интерфейса на третьем уровне

б) Удаление адресации всех протоколов.

Система IOS	`no ip address` (режим конфигурирования интерфейса)

Если интерфейсу присвоен адрес какого-либо протокола, например, IP, необходимо удалить адрес с помощью ключевого слова no и команды, устанавливающей

адресацию. Например, для удаления IP-адреса с интерфейса используется команда **no ip address**.

в) Назначение интерфейса группе каналов.

Система IOS	**channel-group** *number* **mode** {**auto** | **desirable** | **on**} (режим конфигурирования интерфейса)

Для физического интерфейса третьего уровня, который планируется использовать как часть канала, применяется команда **channel-group**. Параметр *number* указывает, с каким интерфейсом канала порта связан данный физический интерфейс. Режимы (**mode**) определяют взаимодействие одной стороны канала с другой стороной. (Более подробная информация по режимам канала приведена в разделе "4.4: порты EtherChannel".)

г) Проверка работоспособности интерфейса.

Системы IOS	**no shutdown** (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является **shutdown** (отключен). Чтобы интерфейс функционировал, следует включить его с помощью команды **no shutdown**.

д) Повторение этапов а–г для всех интерфейсов с одинаковой скоростью, которые будут членами данного канала.

Проверка канала

После настройки канала можно проверить его работоспособность с помощью описанных ниже команд.

Системы IOS	**show etherchannel** *number* **port-channel** **show interfaces** *type number* **etherchannel** (обе команды вводятся в режиме привилегированного пользователя)

При использовании команды **show etherchannel** параметр *number* определяет канал порта или номер группы для канала, состояние которого необходимо проверить. Команда **show interfaces** позволяет указать отдельные члены этого канала и просмотреть EtherChannel-параметры для данных интерфейсов.

Пример конфигурирования функции

В приведенном ниже примере демонстрируется конфигурация интерфейсов Gigabit Ethernet 1/1 и Gigabit Ethernet 2/1 коммутатора Distribution_Switch_A (коммутатор уровня распределения) в качестве канала третьего уровня. Этот интерфейс выполняет функции шлюза для всех клиентов, подключенных к коммутатору Access_Switch_A (коммутатор уровня доступа к сети). На рис. 5.2 проиллюстрирована топология сети для этого примера.

Ниже приводится пример конфигурации коммутатора Distribution_Switch_A.

```
Distribution_Switch_A(config)#interface port-channel 1
Distribution_Switch_A (config-if)#ip address 192.168.10.1
255.255.255.0
```

```
Distribution_Switch_A (config-if)#interface gigabitethernet 1/1
Distribution_Switch_A (config-if)#no switchport
Distribution_Switch_A (config-if)#no ip address
Distribution_Switch_A (config-if)#channel-group 1 mode on
Distribution_Switch_A (config-if)#no shut
Distribution_Switch_A(config-if)#interface gigabitethernet 2/1
Distribution_Switch_A (config-if)#no switchport
Distribution_Switch_A (config-if)#no ip address
Distribution_Switch_A (config-if)#channel-group 1 mode on
Distribution_Switch_A (config-if)#no shut
Distribution_Switch_A (config-if)#end
Distribution_Switch_A #copy running-config startup-config
```

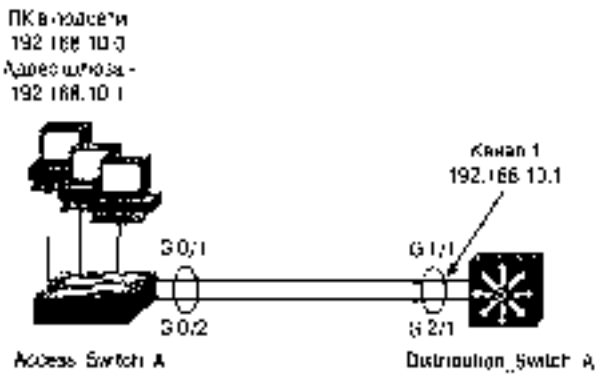


Рис. 5.2. Топология сети для примера конфигурации канала третьего уровня

Пример конфигурации коммутатора Access_Switch_A (3500XL).

```
Access_Switch_A (config)#interface gigabitethernet 0/1
Access_Switch_A (config-if)#port group 1
Access_Switch_A (config)#interface gigabitethernet 0/2
Access_Switch_A (config-if)#port group 1
Access_Switch_A (config-if)#end
Access_Switch_A #copy running-config startup-config
```

5.4: WAN-интерфейсы

- Коммутаторы Catalyst серий 5000 и 6000 предоставляют поддержку WAN-интерфейсов, добавляемых в шасси коммутатора.
- WAN-интерфейсы известны только процессору коммутации третьего уровня и должны конфигурироваться с использованием IOS-интерфейса.
- Коммутатор серии 5000 позволяет добавлять плату RSM/VIP2, которая способна обеспечить поддержку различных *модулей адаптеров портов (Port Adapter Modules — PAM)* для WAN-связи.
- Коммутатор серии 6000 поддерживает плату FlexWAN, которая обеспечивает поддержку различных WAN PAM-модулей для WAN связи.

- В дополнение к плате FlexWAN коммутатор серии 6000 предоставляет различные модули оптических служб, которые могут подключаться к высокоскоростным оптическим сетям.

Конфигурация функции

WAN-интерфейсы позволяют пользователям подключаться к отдаленным службам посредством шасси коммутатора Catalyst. Такие интерфейсы, как правило, функционируют как интерфейсы третьего уровня и для предоставления доступа и связи с платами требуют какого-либо типа процессора коммутации третьего уровня, такого, как RSM и MSFC. Доступ к этим интерфейсам можно получить только из программного обеспечения Cisco IOS, поэтому их можно настраивать лишь с процессора коммутации третьего уровня, или коммутатор должен работать под управлением операционной системы Supervisor IOS.

Внимание!

В текущем разделе представлен краткий обзор конфигурирования некоторых основных параметров для коммутации третьего уровня с использованием WAN-интерфейсов. Источники, откуда вы сможете почерпнуть более подробную информацию по описываемым интерфейсам и WAN-связи, приведены в разделе "Дополнительная литература" в конце этой главы.

В каждом из последующих разделов подробно описаны этапы конфигурирования различных WAN-интерфейсов для обеспечения основной сетевой связи.

Конфигурирование WAN-интерфейсов VIP2

VIP2-интерфейс является компонентом, который позволяет устанавливать PAM-модули в подсистему модуля коммутации маршрутов. VIP2 — компонент комбинированной платы RSM/VIP2, которая занимает два гнезда в шасси коммутатора Catalyst серии 5000. Как часть RSM-модуля, она позволяет устанавливать в коммутатор различные интерфейсы третьего уровня, включая WAN-интерфейсы. Ниже описаны этапы конфигурирования WAN-интерфейсов для PAM-модуля VIP2.

1. Доступ к RSM-модулю.

   WAN-интерфейсы конфигурируются через RSM-модуль. Для того чтобы получить к нему доступ, используется команда `session`

Система COS	`session` *mod* (режим привилегированного пользователя)

   Параметр *mod* определяет номер модуля процессора коммутатора. Для определения расположения RSM-модуля в коммутаторе, в случае, если оно неизвестно, используется команда `show modules`

2. Конфигурирование WAN-интерфейса.

Система IOS	`interface` *type slot/number* (режим глобальной конфигурации)

Для создания и доступа к WAN-интерфейсу используются приведенные выше команды в режиме глобальной конфигурации. Параметр *type* определяет тип WAN-интерфейса. Параметр *bay* указывает, в какой отсек VIP установлен PAM-модуль. Отсеки пронумерованы 0 и 1 слева направо. Параметр *number* определяет номер порта в отсеке.

Внимание!

VIP2-процессор поддерживает несколько различных PAM-модулей. Полный список поддерживаемых им модулей третьего уровня доступен по адресу *www.cisco.com/univercd/cc/td/doc/product/lan/cat5000/relnotes/wan/wan_pa/index.htm*.

3. Назначение адреса интерфейсу.

Система IOS	`ip address address netmask` (режим конфигурирования интерфейса)

Для назначения интерфейсу третьего уровня сетевого адреса используются соответствующие команды. Приведенный выше пример отражает конфигурирование IP-адреса. Параметры для других протоколов приведены в разделе "5.2. Ethernet-интерфейсы третьего уровня", этап 3.

4. Включение интерфейса.

Система IOS	`no shutdown` (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является **shutdown** (отключен). Чтобы обеспечить функционирование интерфейса, следует включить его с помощью команды **no shutdown**.

Конфигурирование FlexWAN-интерфейса

Плата FlexWAN для коммутатора серии 6000 подобна VIP2-плате для коммутатора серии 5000. Этот модуль позволяет устанавливать ограниченное число WAN PAM-модулей для использования процессором коммутации третьего уровня в целях обеспечения WAN-связи. Главное отличие заключается в том, что FlexWAN-плата не связана с какой-либо другой платой (такой, как RSM) и подключается непосредственно к объединительной плате коммутатора. Однако для функционирования FlexWAN-платы требуется предварительная установка в коммутатор платы MSFC. Ниже описаны этапы конфигурирования FlexWAN-портов.

1. Доступ к процессору коммутации третьего уровня (для гибридного режима).

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к данному устройству. Для этого используется команда **session**. При использовании коммутатора, работающего с системой Supervisor IOS, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутатора третьего уровня.

Система COS	`session mod` (режим глобальной конфигурации)

Параметр *mod* указывает номер модуля процессора коммутатора. Чтобы определить расположение процессора в коммутаторе, если оно неизвестно, используется команда `show modules`

Внимание!

Если шасси коммутатора содержит две активные платы MSFC, то FlexWAN-порты доступны только для платы MSFC в гнезде 15 или для активного маршрутизатора, если используется синхронизация конфигурации. Если активный маршрутизатор или плата MSFC в гнезде 15 выходит из строя, необходимо вручную настроить порты на оставшемся маршрутизаторе, прежде чем они смогут функционировать снова.

2. Конфигурирование WAN-интерфейса.

Система IOS	`interface type slot/bay/number` (режим глобальной конфигурации)

Для доступа к WAN-интерфейсу используется приведенная выше команда в режиме глобальной конфигурации. Параметр *type* определяет тип WAN-интерфейса (например, `serial`, `hssi` или `atm`). Параметр *slot* указывает гнездо шасси коммутатора, параметр *bay* является номером отсека на FlexWAN-плате (они нумеруются 0 и 1 слева направо). Параметр *number* определяет номер интерфейса на PAM-модуле (интерфейсы нумеруются, начиная с нуля).

Внимание!

Можно использовать только определенное количество PAM-модулей с модулем FlexWAN. Полный список поддерживаемых адаптеров доступен по адресу *www.cisco.com/univercd/cc/td/doc/product/lan/cat5000/c5protr/flexwan/flex_pa/index.htm*.

3. Назначение адреса интерфейсу.

Система IOS	`ip address` *address netmask* (режим конфигурирования интерфейса)

Для назначения интерфейсу третьего уровня сетевого адреса используются соответствующие команды. Приведенный выше пример отражает конфигурирование IP-адреса. Параметры для других протоколов приведены в разделе "5.2. Ethernet-интерфейсы третьего уровня", этап 3.

2. Включение интерфейса.

Система IOS	`no shutdown` (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является `shutdown` (отключен). Чтобы обеспечить функционирование интерфейса, следует включить его с помощью команды `no shutdown`.

Конфигурирование Gigabit Ethernet WAN-интерфейса

Блок Gigabit Ethernet WAN OSM представляет собой четырехпортовый гигабитный модуль, который может использоваться для подключения к *региональным сетям*

(*Metropolitan Area Network — MAN*) с целью получения гигабитных WAN-служб. Для работы такой платы требуется коммутатор серии 6500 с модулем Supervisor 2 и платой MSFC2. Каждый порт в данном отсеке функционирует в качестве порта третьего уровня и может конфигурироваться только с IP-адресом наряду с различными функциями управления трафиком. Реализация основной конфигурации этих интерфейсов включает в себя следующие этапы.

1. Доступ к процессору коммутации третьего уровня (для гибридного режима).

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к этому устройству. Для этого используется команда **session**. При использовании коммутатора, работающего с системой Supervisor IOS, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутатора третьего уровня.

Система COS	**session** *mod* (режим привилегированного пользователя)

   Параметр *mod* указывает номер модуля процессора коммутатора. Чтобы определить расположение процессора в коммутаторе, если оно неизвестно, используется команда **show modules**.

2. Доступ к интерфейсу.

Система IOS	**interface ge-wan** *slot/number* (режим глобальной конфигурации)

   Для доступа к этому интерфейсу используется указанная команда в режиме глобальной конфигурации. Параметр *type* в этом случае соответствует **ge-wan**, *slot* указывает на номер гнезда в шасси, а параметр *number* соответствует номеру порта.

3. Назначение интерфейсу IP-адреса.

Система IOS	**ip address** *address netmask* (режим конфигурирования интерфейса)

   Эта команда используется для включения на заданном порту функции IP-обработки. Гигабитные порты WAN поддерживают только IP-обработку трафика.

4. Включение интерфейса.

Система IOS	**no shutdown** (режим конфигурирования интерфейса)

   Стандартным состоянием многих интерфейсов третьего уровня является **shutdown** (отключен). Чтобы обеспечить функционирование интерфейса, следует включить его с помощью команды **no shutdown**.

Конфигурирование интерфейса передачи пакетов по сетям SONET

Интерфейсы *передачи пакетов по сетям SONET* (POS-интерфейсы) предоставляют другой метод для подсоединения коммутаторов серии 6500 к высокоскоростным ре-

гиональным сетям. В дополнение к одно-, четырех- или шестнадцатипортовому POS-интерфейсу эти платы также предоставляют четыре Gigabit Ethernet-платы для связи с основной сетью.

Для работы плате необходим коммутатор серии 6500 с модулем Supervisor 2 и платой MSFC 2. Реализация основной конфигурации POS-интерфейсов включает следующие этапы.

1. Доступ к процессору коммутации третьего уровня (для гибридного режима).

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к данному устройству. С этой целью используется команда **session**. Ввод указанной команды не требуется при использовании коммутатора, работающего с системой Supervisor IOS, поскольку в таком случае имеется непосредственное подключение к процессору коммутатора третьего уровня.

Система COS	**session** *mod* (режим привилегированного пользователя)

   Параметр *mod* указывает номер модуля процессора коммутатора. Чтобы определить расположение процессора в коммутаторе, если оно неизвестно, используется команда **show modules**.

2. Доступ к POS-интерфейсу.

Система IOS	**interface pos** *slot/port* (режим глобальной конфигурации)

   Для доступа к POS-интерфейсу используется приведенная выше команда в режиме глобальной конфигурации. Параметр *slot* обозначает гнездо в шасси коммутатора, параметр *port* указывает, какой POS-порт конфигурируется.

3. Указание типа инкапсуляции.

Система IOS	**encapsulation** {**hdlc** | **ppp**} (режим конфигурирования интерфейса)

   Необходимо обеспечить совместимую инкапсуляцию второго уровня между устройствами. При подключении к другому устройству Cisco, как правило, используется протокол HDLC (*High-Level Data Link Control* — *высокоуровневый протокол управления каналом*). При подключении к устройствам других производителей оборудования используется PPP-инкапсуляция.

4. Указание источника синхронизации (*необязательно*).

Система IOS	**clock source** {**line** | **internal**} (режим конфигурирования интерфейса)

   При лабораторном подключении двух коммутаторов (back-to-back) с помощью темного волокна (dark fiber) необходимо настроить один из коммутаторов с параметром **clock source internal**, иначе используется стандартная настройка — **line**.

5. Назначение интерфейсу IP адреса.

   Приведенная выше команда используется для включения функции IP-обработки данного порта.

Система IOS	`ip address` *address netmask* (режим конфигурирования интерфейса)

6. Включение интерфейса.

Система IOS	`no shutdown` (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является `shutdown` (отключен). Чтобы интерфейс функционировал, следует включить его с помощью команды `no shutdown`.

Проверка конфигурации

После настройки WAN-интерфейсов для проверки конфигурации используется привилегиями ниже команда.

Система IOS	`show interface` *type number* (режим привилегированного пользователя)

Пример конфигурирования функции

Ниже приводится пример конфигурации коммутатора серии 5500 (Core_switch_1 коммутатор первого уровня), использующего плату VIP2 с последовательным интерфейсом, подключенным к коммутатору серии 6500 (Core_switch_2) с помощью FlexWAN-интерфейса через сеть Frame Relay. DLCI-идентификатор *(Data-Link Connection Identifier — идентификатор подключения канального уровня)* для коммутатора 5500-й серии равен 110, DLCI для коммутатора серии 6500 — 120. В этом примере коммутатор 6500 работает под управлением операционной системы Supervisor IOS. На рис. 5.3 приведена топология сети, связанная с этим примером.

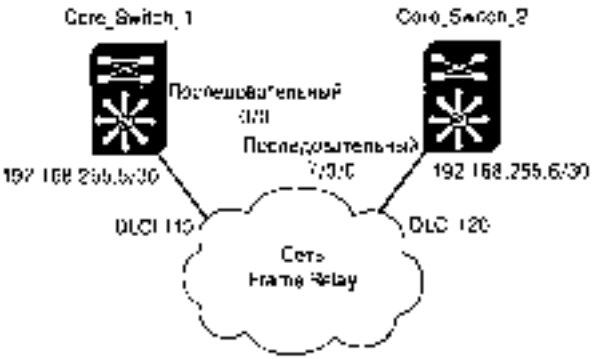


Рис. 5.3. Топология сети для примера конфигурации WAN-интерфейса

Пример конфигурации коммутатора Core_switch_1

```
Core switch 1> (enable)session 4
Core_switch_RSM>enable
Core_switch_RSM#config t
```

```
Core_switch_RSM(config)#interface serial 3/0
Core_switch_RSM(config-if)#encapsulation frame-relay
Core_switch_RSM(config)#interface serial 0/0.110
Core_switch_RSM(config-if)#frame-relay interface-dlci 110
Core_switch_RSM(config-if)#ip address 192.168.255.5 255.255.255.252
Core_switch_RSM(config-if)#no shutdown
Core_switch_RSM(config-if)#end
Core_switch_RSM#copy running-config startup-config
Core_switch_RSM#quit
Core_switch_1> (enable)
```

Пример конфигурации коммутатора Core_switch_2, работающего под управлением операционной системы Supervisor IOS.

```
Core_switch_2>enable
Core_switch_2#config t
Core_switch_2(config)#interface serial 7/0/0
Core_switch_2(config-if)#encapsulation frame-relay
Core_switch_2(config)#interface serial 7/0/0.120
Core_switch_2(config-if)#frame-relay interface-dlci 120
Core_switch_2(config-if)#ip address 192.168.255.6 255.255.255.252
Core_switch_2(config-if)#no shutdown
Core_switch_2(config-if)#end
Core_switch_2#copy running-config startup-config
```

Внимание!

Несмотря на то что в этом примере демонстрируется конфигурирование FlexWAN-модуля с помощью коммутатора 6500 с операционной системой IOS, FlexWAN-модуль может быть настроен с помощью MSFC IOS в гибридном режиме.

5.5: виртуальные интерфейсы

- Виртуальные интерфейсы существуют в конфигурациях, где нет одного физического подключения к широковещательному домену.
- Для коммутаторов с интерфейсами второго уровня широковещательные домены определяются VLAN-сетями.
- VLAN-интерфейс — любой интерфейс третьего уровня для любого члена определенной сети VLAN.
- Для коммутаторов или маршрутизаторов с интерфейсами третьего уровня широковещательные интерфейсы определяются как мостовые группы.
- Для определения маршрута между мостовыми группами и другими широковещательными доменами в качестве интерфейса третьего уровня используется *мостовой виртуальный интерфейс (Bridged Virtual Interface — BVI)*.
- В некоторых случаях физический интерфейс третьего уровня способен поддерживать трафик от нескольких VLAN-сетей.
- Чтобы обеспечить интерфейсы третьего уровня для каждой VLAN-сети на физическом соединении, в качестве интерфейса третьего уровня для членов VLAN-сети конфигурируется какой-либо подынтерфейс.

Конфигурирование VLAN-интерфейса

1. Доступ к процессору коммутации третьего уровня (для гибридного режима)

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к данному устройству. Для этого используется команда **session**. При использовании коммутатора, работающего с системой Supervisor IOS, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутатора третьего уровня.

Система COS	**session** *mod* (режим привилегированного пользователя)

   Параметр *mod* указывает номер модуля процессора коммутатора. Чтобы определить положение процессора в коммутаторе, если оно неизвестно, используется команда **show modules**.

2. Конфигурирование VLAN-интерфейса.

Система IOS	**interface vlan** *number* (режим глобальной конфигурации)

   Приведенная выше команда используется в режиме глобальной конфигурации для создания VLAN-интерфейса и доступа к нему. Этот интерфейс будет находиться в том же широковещательном домене, что и члены VLAN сети с указанным номером. Для того чтобы данный интерфейс стал активным, прежде всего необходимо его присутствие в базе данных VLAN-сетей коммутатора (см. раздел "6.1: конфигурация VLAN-сети").

Внимание!

VLAN-интерфейс на коммутаторе второго уровня, таком, как 2900/3500XL и 2950, не осуществляет коммутацию третьего уровня и не будет функционировать так, как описано в этом разделе. VLAN-интерфейс IOS второго уровня является только административным.

3. Назначение интерфейсу адреса протокола.

Система IOS	**ip address** *address netmask* (режим конфигурирования интерфейса)

   Для назначения интерфейсу третьего уровня сетевого адреса используются соответствующие команды. Приведенный выше пример отражает конфигурирование IP-адреса. Параметры для других протоколов приведены в разделе "5.2 Ethernet-интерфейсы третьего уровня", этап 3.

4. Включение интерфейса.

Система IOS	**no shutdown** (режим конфигурирования интерфейса)

   Стандартным состоянием многих интерфейсов третьего уровня является **shutdown** (отключен). Чтобы интерфейс функционировал, следует включить его с помощью команды **no shutdown**.

Конфигурирование мостовых виртуальных интерфейсов

1. Доступ к процессору коммутации третьего уровня (для гибридного режима).

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к этому устройству. Для этого используется команда `session`. При использовании коммутатора, работающего с системой Supervisor IOS, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутатора третьего уровня.

Система COS	`session` *mod* (режим привилегированного пользователя)

   Параметр *mod* указывает номер модуля процессора коммутатора. Чтобы определить расположение процессора в коммутаторе, если оно неизвестно, используется команда `show modules`.

Внимание!

Несмотря на то что можно сконфигурировать BVI-интерфейс с использованием мостовых групп для виртуальных интерфейсов на модуле RSM или плате RSFC, как правило, такая конфигурация выполняется в случае, если необходимо обеспечить обмен данными второго уровня между устройствами в отдельных VLAN-сетях.

Внимание!

Мостовые группы и BVI-интерфейсы не поддерживаются на интерфейсах, которые с помощью команды `switchport` могут переводиться в режим интерфейсов либо второго, либо третьего уровня.

2. Конфигурирование мостовой группы.

Система IOS	`bridge` *number* `protocol ieee` (режим глобальной конфигурации)

   Для конфигурирования BVI интерфейса прежде всего необходимо создать мостовую группу, в которой для определения маршрута будет использоваться интерфейс BVI. Указанная команда конфигурирует мостовую группу, использующую протокол IEEE. Параметр *number* используется для связывания портов с группой.

3. Включение *интегрированной маршрутизации и установки мостового соединения (Integrated Routing and Bridging — IRB).*

Система IOS	`bridge irb` (режим глобальной конфигурации)

   IRB процесс необходимо активизировать, так как порты в мостовой группе будут маршрутизироваться и связываться с помощью моста.

4. Включение маршрутизации для мостовой группы.

   Стандартно при активизации функции IRB порты, являющиеся членами мостовой группы, не пытаются маршрутизировать пакеты, поскольку они рассматри-

является как местные порты. Если планируется настроить интерфейс третьего уровня для использования портами моста, то необходимо указать, что мостовая группа может осуществлять маршрутизацию. Параметр *number* указывает, для каких мостовых портов осуществляется маршрутизация, а параметр *protocol* определяет, какой протокол (или протоколы) третьего уровня будет маршрутизироваться.

Система IOS	`bridge number route {ip | ipx | appletalk}` (режим глобальной конфигурации)

5. Назначение интерфейсов мостовой группе.

Система IOS	`bridge-group number` (режим конфигурирования интерфейса)

Необходимо назначить мостовой группе все интерфейсы, которые будут находиться в одном широковещательном домене. Клиенты таких интерфейсов будут находиться в одной IP-подсети и использовать BVI в качестве интерфейса третьего уровня или шлюза на выходе из подсети. Параметр *number* соответствует номеру моста на этапах 2 и 4.

6. Конфигурирование BVI-интерфейса.

Система IOS	`interface BVI number` (режим глобальной конфигурации)

Эта команда используется в режиме глобальной конфигурации устройства для создания BVI-интерфейса и доступа к нему. Интерфейс находится в том же широковещательном домене, что и члены мостовой группы, номер которой указан. Параметр *number* указывает, к какой мостовой группе относится данный интерфейс.

7. Назначение интерфейсу адреса протокола.

Система IOS	`ip address address netmask` (режим конфигурирования интерфейса)

Для назначения интерфейсу третьего уровня сетевого адреса используются соответствующие команды. Приведенный выше пример отражает конфигурирование IP-адреса. Параметры для других протоколов приведены в разделе "5.2: Физические интерфейсы третьего уровня", этап 3.

8. Включение интерфейса.

Система IOS	`no shutdown` (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является shutdown (отключен). Чтобы интерфейс функционировал, следует включить его с помощью команды `no shutdown`.

Конфигурирование подынтерфейсов

1. Создание подынтерфейса и доступ к нему.

Эта команда используется в режиме глобальной конфигурации для создания подинтерфейса и доступа к нему. Параметр *type* соответствует типу контроллера интерфейса (например, `fastethernet` и `gigabitethernet`). Также может использоваться тип `port-channel` для соединения с объединением каналов. Параметр *number* определяет расположение или логический номер интерфейса, а параметр *.subnumber* создает логический интерфейс третьего уровня от главного соединения.

Система IOS	`interface` *type number.subnumber* (режим глобальной конфигурации)

2. Определение инкапсуляции и VLAN-сети.

Система IOS	`encapsulation {dot1q | isl}` *vlannumber* `[native]` (режим конфигурирования подынтерфейса)

В режиме конфигурирования подынтерфейса с помощью команды `encapsulation` указывается, какая VLAN-сеть связана с данным подынтерфейсом. Тип (`dot1q` или `isl`) зависит от типа магистрального канала, подключенного к интерфейсу маршрутизатора. Параметр *vlannumber* указывает, какая VLAN-сеть связана с подынтерфейсом, т.е. в каком широковещательном домене этот подынтерфейс будет функционировать в качестве интерфейса третьего уровня.

Только для магистральных каналов dot1q параметр `native` указывает, какая из VLAN-сетей будет для канала собственной, что очень важно, поскольку пакеты собственной VLAN-сети, согласно спецификации 802.1Q не маркируются.

Внимание!

Подынтерфейсы используются в конфигурациях для маршрутизаторов или интерфейсов, подключённых к магистральным каналам. Интерфейсы третьего уровня не используют *динамический магистральный протокол (Dynamic Trunking Protocol — DTP)*, и любой коммутатор, подключенный к таким интерфейсам, должен быть сконфигурирован в режиме включенного магистрального канала.

Внимание!

Интерфейсы на коммутаторах Catalyst серии 4000 и 6000, использующих Supervisor IOS, а также на коммутаторах серии 3550 не поддерживают подынтерфейсы. Вместо них следует использовать магистральный порт и VLAN-интерфейсы, описанные в этом разделе.

4. Назначение интерфейсу адреса протокола.

Система IOS	`ip address` *address netmask* (режим конфигурирования интерфейса)

Для назначения интерфейсу третьего уровня сетевого адреса используются соответствующие команды. Приведенный выше пример отражает конфигурирование IP-адреса. Параметры для других протоколов приведены в разделе "5.2: Ethernet-интерфейсы третьего уровня", этап 3.

4. Включение интерфейса.

Система IOS	`no shutdown` (режим конфигурирования интерфейса)

Стандартным состоянием многих интерфейсов третьего уровня является `shutdown` (отключен). Чтобы интерфейс функционировал, следует включить его с помощью команды `no shutdown`.

Совет

Коммутатор третьего уровня Catalyst 4000 подключается к коммутатору второго уровня посредством двух внутренних гигабитовых интерфейсов. Каждый из них (Gigabit Ethernet 3 и Gigabit Ethernet 4) является интерфейсом третьего уровня и может быть сконфигурирован индивидуально (если требуется только маршрутизация для одной или двух VLAN-сетей) или с подынтерфейсами, как описано в данном разделе. Другой вариант заключается в объединении интерфейсов в создании магистрального канала через канал, который образует подынтерфейсы для канала порта.

Проверка конфигурации

После настройки подынтерфейсов для проверки конфигурации используются приведенные ниже команды.

Система IOS	`show interface type number.subnumber`
	`show vlan [number]`
	(режим привилегированного пользователя)

Пример конфигурирования функции

В этом примере демонстрируется конфигурация коммутатора 2948G-L3, подключенного к коммутатору серии 3550 через магистральный канал 802.1Q, проходящий между портами G49 на коммутаторе 2948G-L3 и G0/1 — на коммутаторе 3550. На обоих коммутаторах сконфигурирован виртуальный интерфейс для сети VLAN 10. На рис. 5.4 показана топология сети, соответствующая этому примеру.

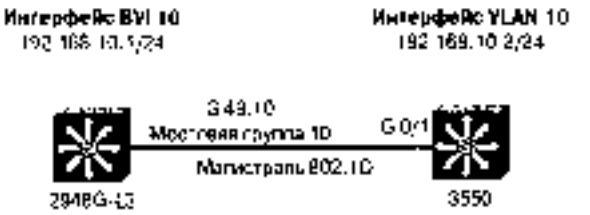


Рис. 5.4. Топология сети для примера конфигурации виртуального интерфейса

Пример конфигурации коммутатора 2948G-L3.

```
2948G-L3 (config)#bridge 10 protocol ieee
2948G-L3 (config)#bridge irb
2948G-L3 (config)#bridge 10 route ip
2948G-L3 (config)#interface gigabitethernet 49.10
2948G-L3 (config-subif)#encapsulation dot1q 10
```

```
2960G-L3 (config-subif)#bridge-group 10
2960G-L3 (config-subif)#no shutdown
2960G-L3 (config-subif)#interface BVI 10
2960G-L3 (config-if)#ip address 192.168.30.3 255.255.255.0
2960G-L3 (config-if)#no shutdown
2960G-L3 (config-if)#end
2960G-L3 #copy running-config startup-config
```

```
3550 #vlan database
3550 (vlan)#vlan 10
3550 (vlan)#exit
3550 #config t
3550 (config)#interface gigabitethernet 0/1
3550 (config-if)#switchport mode trunk
3550 (config-if)#switchport mode on
3550 (config-if)#switchport trunk encapsulation dot1q
3550 (config-if)#interface vlan 10
3550 (config-if)#ip address 192.168.10.2 255.255.255.0
3550 (config-if)#no shutdown
3550 (config-if)#end
3550 #copy running-config startup-config
```

5.6: таблицы маршрутизации

- Чтобы перемещать пакеты между отдельными сетями, процессору коммутации требуется информация о сети получателя
- Сети, подключенные к физическому или виртуальному интерфейсу, соединяют маршруты и автоматически опознаются процессором коммутации
- В процессоре коммутации третьего уровня можно построить статически определенные маршруты путем их записи в конфигурационный файл.
- Одним из наиболее распространенных способов определения и поддержки маршрутов является использование динамических протоколов маршрутизации, таких, как OSPF (*Open Shortest Path First Protocol — протокол первоочередного открытия кратчайшего маршрута*) и EIGRP (*Enhanced Interior Gateway Routing Protocol — усовершенствованный протокол маршрутизации внутренних шлюзов*).

Конфигурация

1. Доступ к процессору коммутации третьего уровня

   Если процессор коммутации представляет собой плату, которая функционирует как подсистема в COS-устройстве, то для конфигурирования необходимо получить доступ к этому устройству. Для этого используется команда **session**. При использовании коммутатора, работающего с системой Supervisor IOS, вводить указанную команду не нужно, поскольку в таком случае он непосредственно подключен к процессору коммутатора третьего уровня.

Система COS	**session** *mod* (режим привилегированного пользователя)

Параметр *mod* указывает номер модуля процессора коммутатора. Чтобы определить расположение процессора в коммутаторе, если оно неизвестно, используется команда `show modules`.

2. Создание связанных маршрутов.

Система IOS	`ip address` *address netmask* (режим конфигурирования интерфейса)

При назначении сетевого адреса интерфейса также создается запись для сети в таблице маршрутизации. В приведенном выше примере конфигурируется IP-адрес, но такая же методика верна и для других протоколов (см. раздел "5.2. Ethernet-интерфейсы третьего уровня").

3. Создание статических маршрутов.

Система IOS	`ip route` *network netmask {nexthop | interface} [admin-distance]* (режим глобальной конфигурации)

В команде указывается статический маршрут для сети с указанной маской. Параметры *nexthop* или *interface* показывают, каким образом связаться с конфигурируемой сетью.

4. Включение динамических маршрутов.

Система IOS	`router` *protocol* (режим глобальной конфигурации) `network` *network* (режим конфигурирования протоколы маршрутизации)

Команда `router` и указание протокола, такого, как RIP (*Routing Information Protocol — протокол маршрутной информации*), OSPF или EIGRP, переводит пользователя в режим конфигурирования маршрутизатора. В этом режиме указываются сети, для которых необходимо запустить данный протокол.

Внимание!

В этом разделе представлен краткий обзор создания и конфигурирования маршрутов. Раздел задуман как памятка, а не всеобъемлющее изложение методики конфигурирования протоколов маршрутизации. Коммутатор третьего уровня при поддержке маршрутов функционирует в точности так же, как и маршрутизатор. Более подробные сведения, касающиеся конфигурирования, приведены в книге *"Полное руководство Cisco по конфигурированию маршрутизаторов"*, изданной Cisco Press. Ресурсы, представляющие более общую информацию по маршрутизации и соответствующим технологиям, перечислены в разделе "Дополнительная литература" настоящей главы.

Проверка маршрутов

После настройки порта для работы в магистральном режиме следует использовать приведенную ниже команду для проверки назначений VLAN-портов.

Система IOS	`show route` (режим привилегированного пользователя)

Параметр `protocol` позволяет просмотреть таблицу маршрутизации для определенного протокола, такого, как IP, IPX и AppleTalk.

Дополнительная литература

Рекомендуемые ниже источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Коммутация третьего уровня (маршрутизация) и обновление маршрутной информации

Radia Perlman. *Interconnections: Bridges, Routers, Switches and Internetworking Protocols*, Sec. ed., Addison-Wesley.

Steve McQuerry. *Interconnecting Cisco Network Devices*. Cisco Press

Jeff Doyle. *Routing TCP/IP*, Vol. I, Cisco Press.

Jeff Doyle, Jennifer DeHaven, *Routing TCP/IP*, Vol. II, Cisco Press.

David Hucaby and Steve McQuerry, *Cisco Field Manual: Router Configuration*, Cisco Press.

WAN-интерфейсы

Замечания по установке и конфигурированию модуля коммутации маршрутов Catalyst VIP2-15 и VIP2-40 (Route Switch Module Catalyst VIP2-15 and VIP2-40 Installation and Configuration Note): `www.cisco.com/univercd/cc/td/doc/product/lan/cat5000/cnfg_nts/rsm/47e0vip2.htm`.

Документация по FlexWAN-модулю семейства Catalyst 6000 и адаптеру порта (Catalyst 6000 Family FlexWAN Module and Port Adapter Documentation): `www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/cfgnotes/flexwan/index.htm`

Замечания по установке и конфигурированию модулей оптических служб (Optical Services Modules Installation and Configuration Note): `www.cisco.com/univercd/cc/td/doc/product/core/cis7600/cfgnotes/osm_inst/index.htm`.

В этой главе...

- **6.1: конфигурация VLAN-сети.** В этом разделе приводится описание методики конфигурирования, создания и настройки VLAN-сетей на коммутаторе.
- **6.2: назначение сетям VLAN портов.** В разделе рассказывается о назначении порта VLAN-сети с помощью статических или динамических методов.
- **6.3: транкинг.** В этом разделе описан метод расширения VLAN-сети за пределы одного коммутатора посредством механизма маркирования.
- **6.4: протокол магистральных каналов VLAN-сетей.** В этом разделе представлено описание фирменного протокола Cisco для поддержки маршрута передачи данных между коммутаторами, соединенными магистральным каналом, и методики отсекания неиспользуемых VLAN-сетей.
- **6.5: протокол GVRP.** В разделе дано объяснение принятого в индустрии стандартного процесса управления трафиком в магистральных каналах, а также описана поддержка VLAN-сетей на коммутаторах для пересылки данных.
- **6.6: частные сети VLAN.** В этом разделе описана функция, которая позволяет осуществлять более дискретное управление трафиком внутри VLAN-сети с использованием структуры частной VLAN-сети.

Глава 6

VLAN-сети и транкинг

6.1: конфигурация VLAN-сети

- *Сети VLAN* — это определенные внутри коммутаторов широковещательные домены, позволяющие внутри устройства второго уровня управлять широковещательными, многоадресатными, одноадресатными рассылками, а также ограниченными рассылками с неизвестным получателем
- VLAN-сети определяются на коммутаторе во внутренней базе данных, которая называется *базой данных протокола магистральных сетей VLAN (VLAN Trunking Protocol database)*. После создания VLAN-сети назначаются порты
- Для идентификации внутри и между коммутаторами VLAN сетям присваивают номера. Коммутаторы Cisco имеют два диапазона VLAN-сетей: *обычный (normal)* и *расширенный (extended)*.
- VLAN-сети обладают множеством настраиваемых параметров, включая имя, тип и состояние сети.
- Некоторые сети являются резервируемыми, а иные могут использоваться для внутренних целей коммутатора.

Создание Ethernet VLAN-сети

VLAN-сети создаются на коммутаторах второго уровня для управления широковещанием и внедрения устройств третьего уровня для обмена данными. Каждая сеть VLAN создается в локальной базе данных используемого коммутатора. Если в коммутаторе отсутствуют сведения о какой-либо VLAN-сети, то он не может передавать трафик для этой сети VLAN через свои порты. VLAN сети создаются по номерам, при этом существуют два диапазона, пригодных для использования VLAN-номеров (обычный диапазон 1-1000 и расширенный — 1025-4096). При создании VLAN-сети можно также начинать ей определенные атрибуты, такие, как имя, тип и операционное состояние. Процесс создания VLAN-сети включает в себя несколько этапов, описание которых приводится ниже.

1. Конфигурирование протокола VTP.

VTP (VLAN Trunking Protocol) — протокол, используемый в коммутаторах Cisco для поддержки между коммутаторами последовательной базы данных, которая в свою очередь используется для поддержки магистральных соединений. Протокол VTP не является необходимым для создания VLAN-сетей, однако он устанавливается корпорацией Cisco для VLAN-конфигурирования между коммутаторами в качестве стандартного средства, облегчающего администрирование VLAN-сетей. Поэтому прежде всего необходимо либо настроить протокол, указав имя VTP-домена, либо отключить его на коммутаторе. Подробно функции протокола VTP описываются в разделе "6.4: протокол магистральных каналов VLAN-сетей".

Внимание!

Для коммутаторов Catalyst 4000 и 6000, использующих систему IOS Supervisor 12.1 (8a) или выше (собственную IOS), также можно настроить VTP-параметры в режиме глобальной конфигурации

- Назначение VTP-имени

Система COS	`set vtp domain` *domain-name*
Система IOS	`vtp domain` *domain-name* (в режиме конфигурирования базы сетей VLAN) или `vtp domain` *domain-name* (режим глобальной конфигурации)

В соответствии со стандартными установками протокол VTP функционирует в серверном режиме и, прежде чем будут созданы какие-либо VLAN-сети, необходимо настроить протокол VTP, задав имя домена. Имя VTP-домена задается с помощью приведенных выше команд. Для IOS-коммутаторов нужно перейти в режим конфигурирования базы данных VLAN (`vlan`) с помощью ввода команды `vlan database` и приглашения привилегированного уровня.

Внимание!

Команда глобальной конфигурации `vtp domain` доступна не на всех коммутаторах, использующих IOS.

- Отключение VTP-синхронизации.

Система COS	`set vtp mode transparent`

Система IOS	`vtp transparent`
	(в режиме конфигурирования базы сетей VLAN)
	или
	`vtp mode transparent`
	(режим глобальной конфигурации)

Альтернативный способ заключается в отключении VTP-синхронизации баз данных. Это позволяет управлять локальной базой данных VTP, не конфигурируя протокол VTP и не используя его. В коммутаторах Catalyst 4000 и 6000, использующих IOS Supervisor 12.1(8a) или выше (собственную IOS), также можно настроить VTP-параметры в режиме глобальной конфигурации.

Внимание!

Команда глобальной конфигурации `vtp mode transparent` доступна не на всех коммутаторах, использующих IOS

- Отключение протокола VTP.

Система COS	`set vtp mode off`
Система IOS	Нет

С выходом версии 7.1.1 операционной системы COS появилась возможность полностью отключить протокол VTP, для чего необходимо ввести команду `set vtp mode off`. После отключения протокола VTP можно администрировать локальную базу данных VTP.

2. Создание VLAN-сети.

VLAN-сети создаются по номерам. Существуют два диапазона VLAN-сетей:

- стандартный диапазон, в который входят сети VLAN с 1 по 1000;
- расширенный диапазон — в него входят сети VLAN с 1025 по 4096.

VLAN сети расширенного диапазона в настоящее время поддерживаются только на коммутаторах, использующих программное обеспечение COS версии 6.1 или более высокой. При создании VLAN-сети необходимо принимать во внимание множество параметров. Многие из них применимы только к VLAN-сетям FDDI и Token Ring. Некоторые сконфигурированные объекты определяют такие параметры, как частные VLAN-сети, которые обсуждаются в другом разделе этой книги. VLAN-сети создаются с помощью команды `set vlan` в COS-устройствах или команды `vlan` в режиме конфигурирования базы данных VLAN IOS-коммутаторов. Для VLAN-сетей технологии Ethernet можно также настроить стандартные параметры, приведенные в табл. 6.1.

Таблица 6.1. Конфигурируемые VLAN-параметры

Параметр	Описание
`name`	Описательное имя VLAN-сети длиной до 32 символов. Если имя не задано, применяется стандартное имя VLAN00XXX, где xxx — номер VLAN-сети
`mtu`	Максимальная допустимая единица передачи данных (размер пакета в байтах), которая может использоваться в данной VLAN-сети. Стандартные значения находятся в диапазоне от 576 до 18190. MTU-размер может быть увеличен до 1600 для Ethernet-сети и превышает это значение для сетей Token Ring и FDDI. Стандартное значение — 1500
`state`	Используется для определения состояния VLAN-сети — активное или приостановленное. В последнем случае работа всех портов приостановлена и передача кадров трафика не разрешена. Стандартное значение — `active`

Внимание!

В конфигурационных командах есть множество других параметров, однако большинство из них связано с настройкой VLAN-сетей FDDI и Token Ring. Ввиду того что эти технологии на сегодняшний день широко используемыми вариантами и описание конфигурации и параметров для VLAN-сетей FDDI и Token Ring в этой книге не рассматриваются. За сведе-

нами по Token Ring и FDDI VLAN-сетям рекомендуется обратиться к документу www.cisco.com/univercd/cc/td/doc/product/lan/cat5000/rel_6_3/config/vlans.htm

а) Создание VLAN-сети из стандартного диапазона.

Система COS	`set vlan vlan-id [name name] [state state] [mtu mtu]`

Система IOS	`vlan vlan-id [name vlan-name] [state {suspend | active}] [mtu mtu-size]`
	(в режиме конфигурирования базы сетей VLAN)
	`vlan vlan-id`
	(режим глобальной конфигурации)
	`vlan vlan-id [mtu mtu-size] [name vlan-name] [state {suspend | active}]`
	(в режиме конфигурирования базы сетей VLAN)

Параметр `vlan-id` определяет VLAN-сеть по номеру. Для COS-коммутатора в функции `vlan-id` можно указать диапазон VLAN-сетей. Однако имя для диапазона сетей задать нельзя, поскольку каждая VLAN-сеть должна иметь уникальное имя. В IOS-коммутаторах VLAN-сети создаются в режиме конфигурирования базы данных VLAN. Коммутаторы Catalyst 6000 и 4000, использующие Supervisor IOS 12.1(8a) и выше, позволяют создавать VLAN-сети в режиме глобальной конфигурации в том случае, если коммутатор работает в прозрачном VTP-режиме. Чтобы сделать это, необходимо с помощью команды `vlan vlan-id` перейти в режим конфигурирования базы данных VLAN, из которого можно управлять параметрами VLAN-сетей.

Внимание!

Модифицировать какие-либо параметры сети VLAN 1 невозможно.

б) Создание VLAN-сети из расширенного диапазона.

В расширенном диапазоне поддерживаются номера сетей до 4096 согласно стандарту 802.1Q. В настоящее время только коммутаторы, использующие операционную систему COS версии 6.1 или выше, способны поддерживать создание и назначение VLAN-сетей в расширенном диапазоне. Протокол VTP для управления VLAN-сетями расширенного диапазона в настоящее время использовать нельзя, и такие сети VLAN не способны передавать данные посредством магистрального ISL-канала (*Inter-Switch Link — межкоммутаторный канал*).

- Включение функции ограничения количества MAC-адресов для протокола распределенного связующего дерева.

Система COS	`set spantree macreduction enable`
Система IOS	Нет

Чтобы позволить коммутаторам использовать расширенный диапазон, необходимо прежде всего включить функцию `spanningtree macreduction`. Она позволяет коммутатору поддерживать большое число экземпляров распределенного связующего дерева с весьма ограниченным количеством

MAC-адресов и вместе с тем соблюдать требования стандарта IEEE 802.1D по идентификаторам мостов для каждого STP-экземпляра.

Внимание!

После того как была создана VLAN-сеть расширенного диапазона, отключить указанную функцию невозможно, не удалив предварительно сеть.

- Создание VLAN-сети расширенного диапазона.

Система COS	`set vlan vlan_id {name name} {state state} {mtu mtu}`
Система IOS	Нет

В нашем случае параметр `vlan-id` соответствует номеру от 1025 до 4096. Номера от 1001 до 1024 зарезервированы Cisco и не могут использоваться.

Внимание!

Для коммутаторов Catalyst серии 6000 с платами FlexWAN система производит внутреннюю идентификацию данных портов, начиная с 1025. Если в системе имеются какие-либо FlexWAN-модули, необходимо зарезервировать достаточное количество VLAN-номеров (начиная с VLAN 1025) для всех FlexWAN-портов, которые нужно установить. В случае, если устанавливаются FlexWAN-порты, использовать сети расширенного диапазона не возможно.

Пример конфигурирования функции

В этом примере необходимо настроить коммутаторы Access_1 и Distribution_1 на поддержку VLAN-сетей 5, 8 и 10 с именами Cameron, Logan и Katie соответственно. Кроме того, коммутатор распределения необходимо настроить на VLAN-сеть номер 2112 с именем Rush.

Ниже приводится пример конфигурации Catalyst OS для коммутатора Distribution_1.

```
Distribution_1 (enable)>set vtp mode transparent
Distribution_1 (enable)>set vlan 5 name Cameron
Distribution_1 (enable)>set vlan 8 name Logan
Distribution_1 (enable)>set vlan 10 name Katie
Distribution_1 (enable)>set spantree macreduction enable
Distribution_1 (enable)>set vlan 2112 name Rush
Distribution_1 (enable)>
```

Пример конфигурации Supervisor IOS для коммутатора Distribution_1.

```
Distribution_1#vlan database
Distribution_1(vlan)#vtp transparent
Distribution_1(vlan)#exit
Distribution_1#conf t
Distribution_1(config)#vlan 5
Distribution_1(config-vlan)# name Cameron
Distribution_1(config-vlan)#vlan 8
Distribution_1(config-vlan)# name Logan
Distribution_1(config-vlan)# vlan 10
Distribution_1(config-vlan)# name Katie
```

```
Distribution_1(config-vlan)# end
Distribution_1 #copy running-config startup config
```

Внимание!

В операционной системе Supervisor IOS VLAN-сети расширенного диапазона, такие, как 2112, не поддерживаются.

Пример конфигурации IOS второго уровня для коммутатора Access_1.

```
Access_1#vlan database
Access_1 (vlan)#vtp transparent
Access_1 (vlan)#vlan 5 name Cameron
Access_1 (vlan)#vlan 8 name Logan
Access_1 (vlan)#vlan 10 name Katie
Access_1 (vlan)#exit
Access_1#copy running-config startup-config
```

6.2: назначение сетям VLAN портов

- VLAN сети назначаются отдельным портам коммутатора.
- Порты могут назначаться отдельной сети VLAN статически или динамически.
- Все порты стандартно назначены сети VLAN 1.
- Порты являются активными, только если они назначены VLAN-сетям, существующим в данном коммутаторе.
- Статическое назначение портов осуществляется администратором и изменяется только при модификации администратором независимо от того, существует VLAN-сеть на коммутаторе или нет.
- Динамические VLAN-сети назначаются портам согласно MAC-адресу устройства, подключенного к порту.
- Для соответствующего функционирования динамической VLAN-сети требуется наличие клиента, сервера и базы данных *сервера правил размещения VLAN-сетей (VLAN Membership Policy Server — VMPS)*.

Конфигурирование статических VLAN-сетей

На коммутаторах Cisco порты назначаются одной сети VLAN. Такие порты обеспечивают соединение для конечных пользователей или узловых устройств, таких, как маршрутизатор и сервер, и называются *портами доступа к сети (access ports)*. Стандартно все устройства назначаются сети VLAN 1, которая называется *стандартной сетью VLAN (default VLAN)*. После создания VLAN-сети можно вручную назначить ей порт, который сможет обмениваться данными только с другими устройствами в ней. Ниже описаны необходимые действия по конфигурированию порта коммутатора для включения в состав определенной VLAN-сети.

1. Статические назначение сетей VLAN.

   Чтобы изменить VLAN-сеть для какого-либо COS-устройства, используется команда `set vlan` с указанием номера VLAN-сети (`number`), а затем порта или

портов, которые необходимо добавить к ней. VLAN-назначения, подобные этому, рассматриваются как статические, поскольку они неизменны до тех пор, пока администратор не изменит VLAN-конфигурацию.

Система COS	`set vlan` *number mod/port*
Система IOS	`interface` *type mod/port* (режим глобальной конфигурации) `switchport access vlan` *number* (режим конфигурирования интерфейса)

Для IOS-устройства прежде всего необходимо выбрать порт (или диапазон портов для интегрированной системы IOS), а затем ввести команду `switchport access vlan` с указанием номера (*number*) VLAN-сети.

Внимание!

Если сеть, которой назначается порт, отсутствует в базе данных, порт будет отключен до тех пор, пока VLAN-сеть не будет создана.

Конфигурирование динамических VLAN-сетей

Хотя статические VLAN-сети являются наиболее общей формой назначения портов VLAN, можно сделать так, чтобы коммутаторы динамически выбирали VLAN-сеть на основании MAC-адреса устройства, подключенного к порту. Чтобы добиться такого режима работы, необходимо наличие файла базы данных VTP, VTP-сервера, коммутатора VTP-клиента к динамического порта. После того как эти компоненты будут сконфигурированы соответствующим образом, динамический порт может выбрать VLAN-сеть на основании какого-либо устройства, подключенного к нему. Конфигурирование динамических VLAN-сетей включает себя описанные ниже этапы.

1. Создание файла базы данных VTP.

   Созданный с помощью текстового редактора, например, WordPad или vi, файл базы данных VTP помещается на VMPS-сервер или RCP-сервер (*Remote Copy Protocol — протокол удаленного копирования*). Файл базы данных VTP состоит из следующих элементов:

   - заголовка, который включает в себя имя VMPS-домена;
   - указания режима VMPS-сервера;
   - имени резервной (fallback) VLAN-сети;
   - списка MAC-адресов, преобразованных во VLAN-имена.

   Общее описание файла базы данных VMPS приводится ниже.

```
vmps domain Switchblock.
vmps mode open
vmps fallback default
vmps no-domain-req deny
!
vmps-mac-addrs
!
!
```

```
address 0001.0307.2943 vlan-name GroupA
address 0050.0491.4960 vlan-name Group2
address 0050.DA6F.1130 vlan-name GroupC
```

Первая строка, которая должна присутствовать в файле базы данных VTP, состоит из букв **vmps**, за которыми следует ключевое слово **domain** и доменное имя. Доменное имя совпадает с именем VTP-домена коммутатора (или коммутаторов), отправляющего VMPS-запросы. (О протоколе VTP подробно рассказывается в разделе "6.4: протокол магистральных каналов VLAN-сетей".) Это имя используется в запросе на информацию VMPS-преобразования. Три последующие строки представляют информацию о необходимом режиме работы VMPS-сервера. Ключевые слова **mode open** указывают, существует ли входящий запрос в списке MAC-адресов или же нет. Коммутатор должен поместить такое устройство как стандартную VLAN-сеть, имя которой указано в следующей строке. Имя **default** соответствует сети VLAN 1. Можно также выбрать режим **closed**. В таком случае работа порта будет приостановлена, если устройство отсутствует в таблице MAC-адресов. Параметр **no-domain-req deny** указывает на то, что информация VLAN-преобразования порта не предоставляется устройствам, отправляющим запросы без доменного имени. Все строки, которые начинаются с восклицательных знаков (!), являются комментариями и игнорируются VMPS-сервером.

Запись **vmps-mac-addrs** указывает на начало секции преобразования MAC-адресов во VLAN-имена. Записи вводятся в формате **address** *address* **vlan-name** *vlan_name*, где адрес задается в шестнадцатеричном точечном представлении, а VLAN-имя — точное имя (включая регистр символов), найденное в базе данных VLAN запрашивающего коммутатора. Когда отправляется запрос, это преобразование возвращается на запрашивающий коммутатор. VLAN-назначение основывается на доменном имени. Если имя в локальном коммутаторе не найдено, назначение не состоится.

2. Конфигурирование VMPS-сервера.

   а) Установка метода загрузки VMPS-сервера (*необязательно*).

Система COS	`set vmps downloadmethod {rcp , tftp}`
Система IOS	**Нет**

   Указывается метод загрузки файла базы данных VMPS-сервера: с помощью протокола `rcp` или `tftp`.

   б) Установка сервера-источника VMPS и имени файла.

Система COS	`set vmps downloadserver ipaddress [filename]`
Система IOS	**Нет**

   Этот этап включает конфигурирование IP-адреса RCP- или TFTP-сервера, а также определение имени файла (*filename*) базы данных VMPS.

   в) Включение VMPS-службы

Система COS	`set vmps state enable`
Система IOS	**Нет**

При включении службы VMPS с сервера в память коммутатора считывается файл. С этого момента служба способна отвечать на запросы от коммутаторов VMPS клиентов. Для проверки функционирования VMPS-сервера используются команды `show vmps`, `show vmps mac`, `show vmps vlan` и `show vmps statistics`.

Внимание!

После того как служба VMPS-сервера была включена и VMPS-информация загружена в память сервера, обращения к файлу базы данных VMPS не производятся. Поэтому при внесении изменений в файл базы данных VMPS необходимо либо отключить, а затем повторно включить службу сервера, либо повторно загрузить файл с помощью команды `download vmps`.

3. Конфигурирование VMPS-клиента.

Система COS	`set vmps server` *ipaddress* `[primary]`
Система IOS	`vmps server` *ipaddress* `primary` (режим глобальной конфигурации)

Любой коммутатор, имеющий динамические порты, считается VMPS-клиентом. Для того чтобы данный коммутатор запрашивал динамическую VLAN информацию у сервера, необходимо сообщить ему адрес сервера. Параметр `primary` используется для указания IP адреса основного VMPS-сервера. Можно указать до трех IP-адресов VMPS-серверов. Для подтверждения серверной конфигурации используются команды `show vmps server` (COS-устройства) и `show vmps` (IOS-устройства).

Внимание!

Если коммутатор настроен в качестве VMPS-сервера и имеет динамические порты, его также следует настроить в качестве клиента (см. этап 3) и указать его собственный IP адрес в качестве адреса сервера.

4. Конфигурирование порта для динамического назначения VLAN-сети.

Система COS	`set port membership` *mod/port* `dynamic`
Система IOS	`switchport access dynamic` (режим конфигурирования интерфейса)

Указанные команды переводят порт в режим динамических сетей VLAN. Прежде чем конфигурировать порт в качестве динамического, необходимо настроить коммутатор как клиент (этап 3). После чего порт назначается локальной VLAN сети данного коммутатора, имя которой совпадает с именем, сопоставленным с MAC-адресом подключенного устройства в базе данных VMPS.

Проверка правильности назначения VLAN-сети

После конфигурирования для проверки правильности назначения VLAN-сетей портам используются приведенные ниже команды.

Внимание!

Команда `show interface status` доступна не на всех коммутаторах, использующих систему IOS.

Система COS	`show port`

Система IOS	`show interface` *type mod/port* `switchport`
	или
	`show interface status`
	(режим привилегированного пользователя)

Пример конфигурирования функции

В этом примере порты коммутаторов *Access_1* и *Distribution_1* получили следующие назначения:

- порты 1 и 2 коммутатора доступа и порты 3/1-48 коммутатора распределения статически назначаются сети VLAN 5;
- порты 3 и 4 коммутатора доступа и порты 4/1-48 коммутатора распределения статически назначаются сети VLAN 8;
- порты 5 и 6 коммутатора доступа и порты 5/1-12 и 5/18-24 коммутатора распределения статически назначаются сети VLAN 10.

Коммутатору `Distribution_1` назначается IP-адрес 10.1.1.1. Этот коммутатор служит в качестве VMPS-сервера и получает от сервера 10.1.1.101 файл с именем *vmpsconfig.txt*, который показан в конце примера.

Порты 13-16 коммутатора доступа и порты 5/13-17 на сервере `Distribution_1` являются динамическими. На рис. 6.1 демонстрируются соединения и назначения, связанные с этим примером.

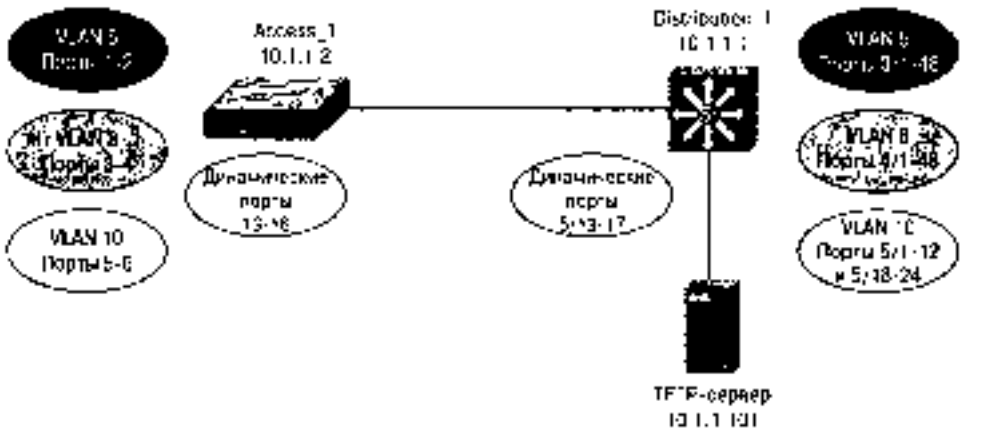


Рис. 6.1. Назначения VLAN-портов на коммутаторах Access_1 и Distribution_1

Пример конфигурации Catalyst OS для коммутатора `Distribution_1`.

```
Distribution_1 (enable)set vlan 5 3/1-48
Distribution_1 (enable)set vlan 8 4/1-48
Distribution_1 (enable)set vlan 10 5/1-12,5/18-24
Distribution_1 (enable)set vmps downloadserver 10.1.1.101
vmpsconfig.txt
Distribution_1 (enable)set int sc0 10.1.1.1/24
Distribution_1 (enable)set vmps enable
```

```
Distribution_1 (enable)set vmps server 10.1.1.1
Distribution_1 (enable)> set port membership 5/13-17 dynamic
```

Пример конфигурации Supervisor IOS для коммутатора Distribution_1

```
Distribution_1(config)#interface range fastethernet 3/1 -48
Distribution_1(config-if)#switchport
Distribution_1(config-if)#switchport mode Access
Distribution_1(config-if)#switchport access vlan 5
Distribution_1(config-if)#no shut
Distribution_1(config)#interface range fastethernet 2/1 -48
Distribution_1(config-if)#switchport
Distribution_1(config-if)#switchport mode access
Distribution_1(config-if)#switchport access vlan 8
Distribution_1(config-if)#no shut
Distribution_1(config)#interface range fastethernet 5/1 -12 ,5/10 -24
Distribution_1(config-if)#switchport
Distribution_1(config-if)#switchport mode access
Distribution_1(config-if)#switchport access vlan 10
Distribution_1(config-if)#no shut
Distribution_1(config-if)#end
Distribution_1 #copy running-config startup-config
```

Внимание!

Для системы Supervisor IOS, работающей на коммутаторе Catalyst 6000 или Catalyst 4000, службы динамических VLAN-сетей в настоящее время не поддерживаются, поэтому эти коммутаторы невозможно настроить с динамическими портами доступа или в качестве VMPS-серверов.

Пример конфигурации системы IOS второго уровня для коммутатора Access_1.

```
Access_1(config)#interface fastethernet 0/1
Access_1(config-if)#switchport access vlan 5
Access_1(config-if)#interface fastethernet 0/2
Access_1(config-if)#switchport access vlan 5
Access_1(config-if)#interface fastethernet 0/3
Access_1(config-if)#switchport access vlan 8
Access_1(config-if)#interface fastethernet 0/4
Access_1(config-if)#switchport access vlan 8
Access_1(config-if)#interface VLAN 1
Access_1(config-if)#ip address 10.1.1.2 255.255.255.0
Access_1(config-if)#vmps server 10.1.1.1
Access_1(config)#interface fastethernet 0/5
Access_1(config-if)#switchport access dynamic
Access_1(config-if)#interface fastethernet 0/6
Access_1(config-if)#switchport access vlan dynamic
Access_1(config-if)# end
Access_1 #copy running-config startup-config
```

Пример файла базы данных VMPS (vmpsconfig.txt)

```
vmps domain SwitchBlock1
vmps mode open
vmps fallback default
vmps no-domain-req allow
```

```
!
vmps-mac-addrs
!
!
address 0001.0387.0940 vlan-name Karie
address 0050.0491.8950 vlan-name Logan
address 0050.DA0F.1234 vlan-name Cameron
```

6.3: транкинг

- VLAN-сети являются локальными в базе данных каждого коммутатора, и информация VLAN-сетей не передается между коммутаторами.
- Магистральные каналы (trunk links) обеспечивают VLAN-идентификацию для фреймов, перемещающихся между коммутаторами.
- В коммутаторах корпорации Cisco имеются два механизма Ethernet-транкинга: протоколы ISL и IEEE 802.1Q.
- Определенные типы коммутаторов способны согласовывать параметры магистральных каналов.
- Магистральные каналы стандартно транспортируют трафик от всех VLAN-сетей к коммутатору и от него, но могут быть настроены на поддержку трафика только определенной VLAN-сети.
- Для того чтобы использовать функции транкинга, на обоих концах канала необходимо настроить магистральные каналы.

Включение транкинга

Магистральные каналы используются для передачи VLAN-информации между коммутаторами. Каждый порт коммутатора Cisco является либо портом доступа к сети (access port), либо магистральным портом (trunk port). Порты доступа принадлежат одной VLAN-сети и не обеспечивают какого-либо идентификационного маркирования фреймов, которые передаются между коммутаторами. Порт доступа также транспортирует трафик, следующий только от той VLAN-сети, которая назначена данному порту. Магистральный порт стандартно является членом всех VLAN-сетей, существующих на коммутаторе, и транспортирует трафик для всех этих сетей между коммутаторами. Чтобы различать потоки трафика, магистральный порт с помощью специальных меток (tags) маркирует фреймы по мере того, как они передаются между коммутаторами. Транкинг (trunking) — функция, которую необходимо включать на обоих концах канала. Например, если два коммутатора соединены вместе, их порты должны быть сконфигурированы для транкинга и оба коммутатора необходимо настроить на один и тот же механизм маркирования (ISL или 802.1Q).

Для включения функции транкинга между коммутаторами необходимо выполнить описанную ниже последовательность действий.

1. Включение транкинга на порту.

   a) Включение магистрального канала (trunk)

Система COS	`set trunk mod/port [auto | desirable | on | nonegotiate | off]`

Система IOS	`interface` *type mod/port* (режим глобальной конфигурации) `switchport mode dynamic [auto	desirable]` `switchport mode trunk` `switchport nonegotiate` (режим конфигурирования интерфейса)

Основным способом конфигурирования магистрального канала является использование параметра `on`. Параметр `on` включает магистральный канал, а также требует указания механизма маркирования для данного магистрального канала. Для IOS-устройств команда `switchport mode trunk` эквивалентна команде `set trunk` *mod/port* `on`. Если указывается параметр `on`, необходимо также указать и механизм маркирования (см. этап 7,б).

Внимание!

Некоторые IOS-коммутаторы не поддерживают динамический магистральный протокол (Dynamic Trunking Protocol). В таких коммутаторах единственной командой, которую можно использовать для конфигурирования транкинга, является команда `switchport mode trunk`, по существу включающая функцию транкинга.

Во многих коммутаторах Cisco зашействован механизм автоматического транкинга, который называется *динамическим магистральным протоколом (Dynamic Trunking Protocol — DTP)*. Этот протокол позволяет динамически создавать магистральный канал между двумя коммутаторами. Все COS-коммутаторы и коммутаторы с интегрированной IOS могут использовать протокол DTP для формирования магистрального канала. Магистральный канал, использующий DTP, конфигурируется с помощью параметров `auto`, `desirable` и `on` на COS-устройствах и параметров `dynamic auto`, `dynamic desirable` и `trunk` на устройствах с операционной системой IOS. Если один из сегмент канала настроен на работу в магистральном режиме и отправляет DTP-сигналы, то другая сторона при точном совпадении параметров переходит в магистральный режим динамически.

Если требуется включить транкинг и не отправлять DTP сигналы, для коммутаторов, поддерживающих эту функцию, используется параметр `nonegotiate`. Если необходимо полностью отключить транкинг, используется параметр `off` для COS-коммутатора или же `switchport mode trunk` на IOS-коммутаторе.

В табл. 6.2 приведены DTP-сигналы и характеристики каждого режима.

Совет

Необходимо помнить о том, что не все коммутаторы поддерживают протокол DTP и без вмешательства пользователя могут не устанавливать магистральный канал. Кроме того, следует помнить, что протокол DTP не предоставляет никаких преимуществ при магистральном соединении с устройствами других производителей оборудования. Поэтому, если протокол DTP не поддерживается для устранения каких-либо непроизводительных затрат, связанных с ним, рекомендуется использовать параметр `nonegotiate`.

Внимание!

При включении транкинга невозможно указать диапазон портов.

Таблица 6.2. Характеристики режимов транкинга

Режим транкинга	Характеристики
COS = on IOS – mode trunk	Для этих каналов функция транкинга включена. В каналах такого типа отправляются DTP-сигналы, которые пытаются инициировать создание магистрального канала с другой стороной. В таком режиме формируется магистральный канал с другими портами в состоянии on, auto или desirable, которые используют протокол DTP. Порт в режиме on всегда маркирует отправляемые с него фреймы
COS = desirable IOS = mode dynamic desirable	Эти каналы готовы к формированию магистральных каналов и отправляют DTP-сигналы, которые пытаются инициировать создание магистрального канала. Канал такого типа станет магистральным, только если получит с другой стороны ответ на DTP-сигнал. При этом формируется магистральный канал с другим портом в состоянии on, auto или desirable, использующим протокол DTP. Этот режим является стандартным для коммутатора серии 6000 с операционной системой Supervisor IOS
COS = auto IOS = mode dynamic auto	Эти каналы станут магистральными, только если получат DTP-сигнал от уже существующего магистрального канала или канала, готового стать магистральным. В таком режиме магистральный канал формируется только с портами в состоянии on или desirable. Стандартный режим для COS-коммутаторов
COS = nonegotiate IOS = mode nonegotiate	В этом режиме включается транкинг и отключается протокол DTP. Такие порты станут магистральными только с портами в режиме on или nonegotiate
COS = off IOS = no switchport mode trunk	Этот параметр отключает магистральные и DTP-функции. Режим рекомендуется для любых портов доступа, поскольку предотвращает динамическое создание магистральных каналов

Внимание!

Коммутаторы Cisco 2950 и 3500XL не поддерживают протокол DTP и всегда работают в режиме, подобном nonegotiate. Если включить транкинг на одном из таких устройств, то согласование с другим концом не состоится. Кроме того, в этом случае требуется, чтобы другой канал был сконфигурирован в режиме on или nonegotiate.

б) Указание метода инкапсуляции.

Система COS	`set trunk mod/port [negotiate | isl | dot1Q]`

Система IOS	`interface type mod/port`
	(режим глобальной конфигурации)
	`switchport trunk encapsulation [negotiate | isl | dot1Q]`
	(режим конфигурирования интерфейса)

Дополнительным параметром при выборе магистрального канала является метод инкапсуляции. Для IOS-коммутаторов второго уровня, таких, как 2900XL или 3500XL, стандартным методом инкапсуляции является isl. Изменить стандартную настройку можно при помощи команды **switchport trunk encapsulation**. Для COS-коммутаторов или коммутаторов с интегрированной системой IOS стандартной инкапсуляцией является **negotiate**. В такой конфигурации между магистральными портами отправляются сигналы для выбора

метода инкапсуляции (метод ISL является более предпочтительным, чем 802.1Q). Параметр `negotiate` применим только для транкинговых режимов `auto` и `desirable`. Если в качестве режима выбирается параметр `on` или необходимо задействовать определенный метод, или другая сторона магистрального канала неспособна согласовать тип транкинга, то для указания метода инкапсуляции необходимо выбрать параметр `isl` или `dot1q`.

Внимание!

Не все коммутаторы позволяют согласовывать настройки инкапсуляции магистрального канала. Магистральные каналы таких коммутаторов, как 2900XL и 3500XL, стандартно настроены на `isl`-инкапсуляцию и для изменения типа инкапсуляции необходимо использовать команду `switchport trunk encapsulation`. Коммутатор 2950 и некоторые коммутаторы серии 4000 поддерживают только 802.1Q-транкинг и не предоставляют способ изменить тип магистрального канала.

в) Определение собственной (native) сети VLAN *(необязательно)*.

Система COS	`set vlan` *number mod/port*
Система IOS	`interface` *type mod/port* (режим глобальной конфигурации) `switchport trunk native vlan` *number* (режим конфигурирования интерфейса)

Для коммутаторов, использующих в качестве механизма транкинга протокол 802.1Q, собственные сети VLAN каждого порта по магистральному каналу должны совпадать. Стандартно все COS-порты находятся в сети VLAN 1; в качестве собственной сети на IOS-устройствах также настраивается сеть VLAN 1, поэтому собственные VLAN-сети совпадают. Если требуется изменить собственную сеть VLAN, используются команды `set vlan` для COS-коммутаторов и `switchport trunk native vlan` — для IOS-коммутаторов, задающие собственную сеть VLAN. Следует помнить о том, что собственная VLAN-сеть должна совпадать на обоих концах магистрального канала с инкапсуляцией 802.1Q, в противном случае канал не будет работать. Если есть несовпадение собственной VLAN-сети, то *протокол распределенного связующего дерева (Spanning Tree Protocol — STP)* переводит порт в несогласованное состояние, и передача через него осуществляться не будет.

Внимание!

Протокол обнаружения устройств Cisco (CDP) версии 2 передает информацию собственной VLAN-сети между коммутаторами Cisco. Если имеется несоответствие собственной VLAN-сети, то на консоли будут отображаться сообщения об ошибках протокола CDP.

Установка VLAN-сетей в магистральный канал

Стандартно магистральные каналы поддерживают все существующие на коммутаторе сети VLAN, поскольку на магистральном канале все сети являются активными. Трафик для определенной VLAN-сети транспортируется по магистральным каналам до тех пор, пока сеть VLAN существует в локальной базе данных коммутатора. Суще-

ствует возможность выборочно удалять и добавлять VLAN-сети в магистральный канал. Для определения того, какие VLAN-сети необходимо добавить или удалить с магистрального канала, используются приведенные ниже команды.

1. Удаление VLAN-сетей из магистрального канала вручную (*необязательно*)

Система COS	`clear trunk mod/port vlanlist`

Система IOS	`interface type mod/port`
	(режим глобальной конфигурации)
	`switchport trunk allowed vlan remove vlanlist`
	(режим конфигурирования интерфейса)

Указанным в параметре `vlanlist` VLAN-сетям будет запрещено транспортировать данные через магистральный канал до тех пор, пока они не будут снова добавлены в магистральный канал с помощью команды `set trunk mod/port vlanlist` или `switchport trunk allowed vlan add vlanlist`.

Проверка магистральных каналов

После конфигурирования портов для транкинга с целью проверки VLAN-назначений портов используются следующие команды.

Система COS	`show trunk [mod] [mod/port]`

Система IOS	`show interface type mod/port switchport`
	или
	`show interfaces trunk`
	или
	`show interface [mod] [interface_id] trunk`
	(режим привилегированного пользователя)

Внимание!

Команды `show interfaces trunk` и `show interface [mod] [interface_id] trunk` доступны не на всех коммутаторах, использующих операционную систему IOS.

Пример конфигурирования функции

В этом примере коммутаторы `Access_1`, `Distribution_1` и `Core_1` подключены согласно схеме на рис. 6.2. Между коммутаторами `Access_1` и `Distribution_1` 802.1Q-транкинг сконфигурирован в режиме "включен" (on). В канале коммутатора `Distribution_1`, подключенном к основе сети, протокол ISL настраивается в режиме **desirable**. Основа сети (core) настраивается на автоматический режим транкинга и согласование инкапсуляции. Магистральный канал, созданный между коммутаторами доступа, настроен только как магистральный канал для сетей VLAN 5, 8 и 10. Магистральный канал между коммутаторами `Distribution_1` и `Core_1` настроен только на транспортировку данных сетей VLAN 1 и VLAN 10.

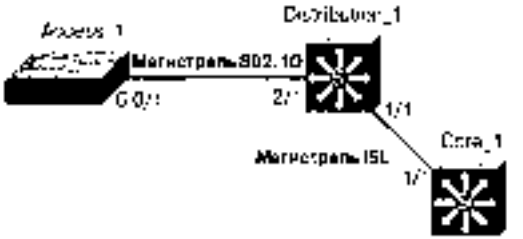


Рис. 6.2. Сетевая диаграмма конфигурации линий магистральных каналов на коммутаторах Access_1, Distribution_1 и Core_1

Пример конфигурации Catalyst OS для коммутатора Distribution_1.

```
Distribution_1 (enable)>clear trunk 1/1 2-1001
Distribution_1 (enable)>set trunk 1/1 desirable isl 10
Distribution_1 (enable)>clear trunk 2/1 2-1001
Distribution_1 (enable)>set trunk 2/1 on dot1q 5,8,10
```

Пример конфигурации Catalyst OS для коммутатора Core_1

```
Core_1 (enable)>clear trunk 1/1 2-1001
Core_1 (enable)>set trunk 1/1 10
```

Пример конфигурации Supervisor IOS для коммутатора Core_1.

```
Core_1 (config)#interface gigabitethernet 1/1
Core_1 (config-if)#switchport encapsulation negotiate
Core_1 (config-if)#switchport mode dynamic auto
Core_1 (config-if)#switchport trunk allowed vlan remove 2-1001
Core_1 (config-if)#switchport trunk allowed vlan add 10
Core_1 (config-if)#end
Core_1#copy running-config startup-config
```

Пример конфигурации IOS второго уровня для коммутатора Access_1

```
Access_1 (config)#interface gigabitethernet 0/1
Access_1 (config-if)#switchport mode trunk
Access_1 (config-if)#switchport trunk encapsulation dot1q
Access_1 (config-if)#switchport trunk allowed vlan remove 2-1001
Access_1 (config-if)#switchport trunk allowed vlan add 5,8,10
Access_1 (config-if)#end
Access_1#copy running-config startup-config
```

6.4: протокол магистральных каналов VLAN-сетей

- Протокол VTP (VLAN Trunking Protocol — протокол магистральных каналов VLAN-сетей) направляет сообщения между коммутаторами, соединенными посредством магистральных каналов, с целью поддержки VLAN-сетей на коммутаторах для соответствующей работы магистрального канала.
- Протокол VTP является фирменным методом корпорации Cisco для управления VLAN сетями между коммутаторами и выполняется при любом типе механизма транкинга.
- Коммутаторы внутри общего VTP-домена обмениваются VTP-сообщениями.

- Прежде чем можно будет создать какую-либо VLAN сеть, необходимо определить VTP-домены или отключить протокол VTP.
- Обмен VTP-информацией может контролироваться при помощи паролей.
- Протокол VTP управляет только ограниченным диапазоном VLAN-сетей: со 2 по 1002.
- Протокол VTP позволяет коммутаторам синхронизировать VLAN-сети на основании номера ревизии конфигурации (configuration revision number).
- Коммутаторы могут функционировать в одном из трех VTP-режимов: серверном, прозрачном или клиентском.
- С помощью протокола VTP можно "отсекать" данные VLAN сети от магистральных каналов.

Включение протокола VTP

Протокол VTP (VLAN Trunking Protocol — протокол магистральных каналов VLAN-сетей) предназначен для обеспечения существования VLAN-сети в локальной базе данных VLAN-коммутаторов на магистральном маршруте. В дополнение к этому протокол VTP способен синхронизировать настройки имен и может применяться для обсекания VLAN-сетей от магистральных каналов, которые направлены к коммутаторам, не имеющим активных портов в сетях VLAN.

Управление и конфигурирование протокола VTP включает в себя перечисленные ниже этапы.

1. Активация протокола VTP на коммутаторе.
   a) Указание имени VTP-домена.

Система COS	`set vtp domain` *name*
Система IOS	`vlan database` (режим привилегированного пользователя) `vtp domain` *name* (режим конфигурирования базы виртуальных сетей) или `vtp domain` *name* (режим глобальной конфигурации)

Стандартно протокол VTP функционирует в серверном режиме (server mode), который позволяет управлять VLAN-сетями в локальной базе данных коммутатора и использовать информацию из базы данных для синхронизации с другими коммутаторами. Для конфигурирования протокола VTP необходимо задать имя. После включения транкинга это имя распространяется среди коммутаторов, для которых имя не было задано. Однако, если планируется построение имен на коммутаторах, необходимо помнить о том, что VTP-имена чувствительны к регистру символов и должны точно совпадать. Коммутаторы, имеющие различные VTP-имена, VLAN-информацией не обмениваются.

Внимание!

Команда `vtp domain` режима глобальной конфигурации поддерживается не всеми коммутаторами, использующими операционную систему IOS.

Внимание!

Понятие VTP-имен используется только в контексте синхронизации баз данных VTP. Имена VTP-доменов не разделяют широковещательные домены. Если сеть VLAN 20 существует на двух соединенных магистральным каналом коммутаторах с различными именами VTP-доменов, то она представляет собой один широковещательный домен.

б) Включение магистрального канала.

Система COS	`set trunk` *mod/port* `{auto	desirable	on	nonegotiate	off}`
Система IOS	`interface` *type mod/port* (режим глобальной конфигурации) `switchport mode dynamic {auto	desirable}` `switchport mode trunk` `switchport nonegotiate` (режим конфигурирования интерфейса)

VTP-информация передается только через магистральные каналы. Если магистральный канал не включен, коммутаторы VLAN-информацией не обмениваются. Более подробные сведения по транкингу приведены в разделе "6.3. Транкинг".

Внимание!

Некоторые IOS-коммутаторы не поддерживают протокол DTP. Для таких коммутаторов единственной применимой для настройки транкинга командой является `switchport mode trunk`, которая по существу включает данную функцию.

Установка VTP-паролей

Стандартно обновления VTP-информации не содержат паролей, и любой коммутатор, не имеющий VTP-имени, подключается к VTP-домену в случае, если функция транкинга включена. Также любой коммутатор, имеющий то же имя VTP-домена, подключается к домену и обменивается VTP-информацией. При этом какой-либо коммутатор в сети может несанкционированно управлять базой данных VLAN-сетей на каждом коммутаторе. Чтобы предотвратить подобную ситуацию, необходимо установить VTP-пароль на коммутаторах, которым необходимо обмениваться данными.

1. Установка VTP-пароля (*необязательно*)

Пароль вводится на каждом коммутаторе, который будет входить в состав VTP-домена. Пароли чувствительны к регистру символов и должны точно совпадать. Если требуется удалить их, используется команда `set vtp passwd 0` на COS-устройствах и команда `no vtp password` в режиме конфигурации базы данных VLAN на устройствах с IOS.

Система COS	`set vtp passwd` *password*
Система IOS	`vlan database` (режим привилегированного пользователя)

```
vtp password password
(режим конфигурирования базы данных сетей VLAN)
или
vtp password password
(режим глобальной конфигурации)
```

Внимание!

Длина пароля для протокола VTP должна быть в пределах от 8 до 32 символов.

Внимание!

Команда `vtp password` поддерживается не на всех коммутаторах, использующих IOS.

Изменение VTP-режимов

Протокол VTP функционирует в одном из трех режимов: серверном, клиентском и прозрачном. Режимы определяют, каким образом передается VTP-информация, как синхронизируются базы данных VLAN и можно ли управлять VLAN-сетями данного коммутатора.

1. Установка VTP режима (*начало описания*).

Система COS	`set vtp mode [server | client | transparent]`

Система IOS	`vlan database`
	(режим привилегированного пользователя)
	`vtp [server | client | transparent]`
	(режим конфигурирования базы данных сетей VLAN)
	или
	`vtp mode [server | client | transparent]`
	(режим глобальной конфигурации)

Стандартно коммутаторы Cisco работают в серверном режиме протокола VTP. VTP-сервер позволяет создавать, удалять и модифицировать VLAN-сети в локальной базе данных VLAN. После внесения изменения они распространяются среди всех остальных коммутаторов VTP-домена, которые работают в серверном или клиентском режиме. Сервер также принимает изменения базы данных VLAN от других коммутаторов домена. Протокол VTP может использоваться в режиме клиента. Коммутаторы в режиме клиента не могут создавать, модифицировать и удалять VLAN-сети из локальной базы данных. Вместо этого для обновления информации о VLAN-сетях они обращаются к другим коммутаторам в домене. Клиенты синхронизируют свои базы данных, но не сохраняют VLAN-информацию и теряют ее при отключении питания. Кроме того, клиенты распространяют сведения о своих базах данных и передают VLAN-информацию другим коммутаторам. Прозрачный режим во многом подобен серверному: в коммутаторы можно создавать, удалять и модифицировать VLAN-сети в локальной базе данных. Отличие заключается в том, что эти изменения не распространяются на другие коммутаторы. В дополнение к этому локальная база данных VLAN не принимает изменений от других коммутаторов. Коммутаторы в прозрачном VTP-режиме пересылают или ретранслируют информацию между другими комму-

транзит и серверном или клиентском режимах. Коммутатор, работающий в прозрачном режиме, не требует настройки имени VTP-домена.

Внимание!

Команда режима глобальной конфигурации `vtp mode` поддерживается не всеми коммутаторами, использующими IOS.

Внимание!

Что касается операционной системы COS версии 7.1(1), то в ней корпорация Cisco представила режим отключенного VTP (`set vtp mode off`), который подобен прозрачному режиму, но не позволяет коммутаторам ретранслировать VTP-информацию между ними. Эта команда полезна в том случае, когда отправлять или передавать VTP-обновления нежелательно, например, если используется транкинг с коммутаторами других производителей оборудования или если для управления базой данных используется динамическое создание VLAN-сетей с помощью общего протокола регистрации сетей VLAN (*Generic VLAN Registration Protocol — GVRP*).

Включение VTP-отсекания

При стандартных настройках все VLAN-сети, существующие на коммутаторе, активно задействуют магистральный канал. Как отмечалось в разделе "802.1Q транкинг", можно вручную удалять VLAN-сети из магистрального канала, а затем добавлять их. VTP-отсекание позволяет коммутатору не передавать трафик пользовательских широковещательных VLAN-сетей. Эта функция динамически отсекает нежелательный трафик от магистральных каналов. Если вновь потребуется передавать трафик определенной VLAN-сети, то протокол VTP динамически добавит ее к магистральному каналу.

Внимание!

Динамическое отсекание удаляет из магистрального канала только нежелательный пользовательский трафик, но не препятствует прохождению через канал каких-либо управляющих фреймов, таких, как сообщения протокола STP.

1. Включение VTP-отсекания (*необязательно*)

   а) Включение отсекания.

Система COS	`set vtp pruning enable`

Система IOS	`vlan database`
	(режим привилегированного пользователя)
	`vtp pruning`
	(режим конфигурирования базы данных сетей VLAN)

После того как на одном VTP-сервере в домене включается VTP-отсекание, все остальные коммутаторы домена также активизируют эту функцию. VTP-отсекание можно включить только на коммутаторах, поддерживающих вторую версию протокола VTP, поэтому прежде чем включать функцию, необходимо обеспечить поддержку второй версии VTP на всех коммутаторах.

Внимание!

Коммутатор должен поддерживать VTP версии 2, однако для включения отсекания на коммутаторе вторую версию протокола активизировать необязательно.

6) Указание отсекаемых VLAN-сетей *(необязательно)*.

Система COS	`clear vtp pruneeligible` *vlanlist*
Система IOS	`interface` *type mod/port* (режим глобальной конфигурации) `switchport trunk pruning vlan remove` *vlanlist* (режим конфигурирования интерфейса)

Стандартно все VLAN-сети на магистральном канале можно использовать для отсекания. С помощью этой команды можно удалить VLAN-сеть из списка отсекаемых (eligible list). После того как VLAN-сеть была удалена из списка отсекаемых, она не может быть отсечена протоколом VTP. Чтобы вернуть VLAN-сеть в список, в COS-коммутаторах используется команда `set vtp pruneeligible` *vlanlist*, команда `switchport trunk pruning vlan add` *vlanlist* в IOS-устройствах.

Изменение версии протокола VTP

Протокол VTP поддерживает две версии. Стандартно все коммутаторы функционируют в режиме первой версии VTP, но большинство из них способно поддерживать режим версии 2.

1. Включение второй версии протокола VTP *(необязательно)*.

Система COS	`set vtp v2 enable`

Система IOS	`vlan database`
	(режим привилегированного пользователя)
	`vtp v2-mode`
	(режим конфигурирования базы данных сетей VLAN)
	или
	`vtp version 2`
	(режим глобальной конфигурации)

Вторая версия протокола VTP стандартно отключена. После ее включения на одном коммутаторе все остальные коммутаторы должны также начинать функционировать в режиме второй версии.

Внимание!

Команда режима глобальной конфигурации `vtp version 2` поддерживается не всеми коммутаторами, использующими операционную систему IOS.

Вторая версия протокола VTP предоставляет ряд параметров поддержки, которые перечислены ниже, недоступных в первой версии.

- **Поддержка нераспознанных записей TLV (unrecognized type-length-value support).** VTP сервер или клиент распространяет конфигурационные изменения по всем своим

магистральным каналам, даже если не способны распознать TLV-значения. Нераспознанные TLV сохраняются в энергонезависимой памяти (NVRAM).

- **Прозрачный режим, зависящий от версии.** В первой версии протокола VTP коммутатор, работающий в прозрачном режиме, проверяет VTP-сообщения для определения доменного имени и версии и передает сообщение только при условии, что версия и доменное имя совпадают. Проверку в программном обеспечении блока Supervisor поддерживается только один домен, протокол VTP версии 2 передает свои сообщения в прозрачном режиме без проверки версии.
- **Проверка достоверности (consistency check).** Во второй версии протокола VTP проверка достоверности параметров VLAN (таких, как VLAN-имена и значения) осуществляется только тогда, когда посредством *интерфейса командной строки (Command-Line Interface — CLI)* или *простого протокола управления сетью (Simple Network Management Protocol — SNMP)* вводится новая информация. Проверки достоверности не производятся, когда новая информация извлекается из VTP-сообщения или считывается из NVRAM-памяти. Если сводка по полученному VTP-сообщению корректна, то информация принимается без проверки достоверности.

Проверка функционирования протокола VTP

После конфигурирования протокола VTP для проверки VLAN-назначений портов используются следующие команды:

Система COS	`show vtp domain`
Система IOS	`show vtp status` (режим привилегированного пользователя)

Пример конфигурирования функции

В этом примере коммутаторы `Access_1`, `Distribution_1` и `Distribution_2` назначаются домену с именем GO-CATS. На рис. 6.3 показано, что коммутатор `Access_1` работает в клиентском режиме VTP и соединен с коммутатором `Distribution_1` при помощи магистрального канала с инкапсуляцией 802.1Q. Коммутатор `Distribution_1` настраивается в режиме VTP-сервера и соединяется посредством магистрального ISL-канала с коммутатором `Core_1`, который работает в прозрачном режиме и подключен с помощью ISL-канала к коммутатору `Distribution_2`, работающему также в серверном режиме. В домене включено VTP-отсекание, и все коммутаторы настроены таким образом, что сеть VLAN 10 не является отсекаемой на магистральных каналах. Поскольку протокол VTP выполняется в магистральных каналах, нет необходимости конфигурировать имя VTP-домена на коммутаторе `Distribution_2` или `Access_1`. Кроме того, не требуется конфигурировать функцию отсекания на каждом коммутаторе, ее настройки также распространяются посредством протокола VTP.

Пример конфигурации Catalyst OS для коммутатора `Core_1`.

```
Core_1 (enable)>set vtp mode transparent
Core_1 (enable)>set trunk 1/1 on isl
Core_1 (enable)>set trunk 1/2 on isl
Core_1 (enable)>
```

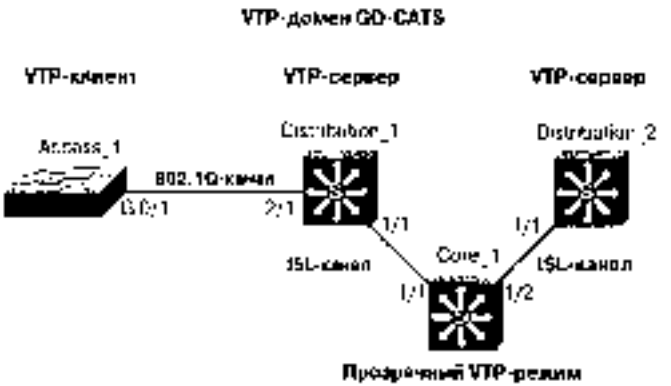


Рис. 6.3. Сетевая диаграмма для VTP-конфигурации, коммутаторов Access_1, Distribution_1, Distribution_2 и Core_1

Пример конфигурации системы Catalyst OS для коммутатора `Distribution_1`

```
Distribution_1 (enable)>set vtp domain GO-CATS
Distribution_1 (enable)>set trunk 1/1 on isl
Distribution_1 (enable)>set trunk 2/1 on dot1Q
Distribution_1 (enable)>set vtp pruning enable
Distribution_1 (enable)>clear vtp pruneeligible 10
```

Пример конфигурации Catalyst OS для коммутатора `Distribution_2`.

```
Distribution_2 (enable)>set trunk 1/1 on isl
Distribution_2 (enable)>clear vtp pruneeligible 10
```

Пример конфигурации IOS второго уровня для коммутатора `Access_1`

```
Access_1#vlan database
Access_1 (vlan)#vtp client
Access_1 (vlan)#exit
Access_1 #config t
Access_1 (config)#interface gigabitethernet 0/1
Access_1 (config-if)#switchport mode trunk
Access_1 (config-if)#switchport trunk encapsulation dot1Q
Access_1 (config-if)#switchport trunk pruning vlan remove 10
Access_1 (config-if)#end
Access_1#copy running-config startup-config
```

6.5: протокол GVRP

- *GVRP (GARP VLAN Registration Protocol — протокол VLAN-регистрации, основанной на общем протоколе атрибутивной регистрации (Generic Attribute Registration Protocol — GVRP))* является приложением, определенным в стандарте IEEE 802.1Q, которое позволяет управлять VLAN-сетями.
- Протокол GVRP взаимодействует только на магистральных каналах 802.1Q.
- Протокол GVRP отсекает магистральные каналы таким образом, что только активные VLAN-сети направляют данные через магистральные соединения.

- Прежде чем добавить VLAN-сеть к магистральному каналу, протокол GVRP ожидает от коммутатора сообщения о подключении.
- Обновления и таймеры удержания протокола GVRP могут быть изменены.
- Чтобы контролировать отсекание VLAN-сетей, используются различные режимы работы GVRP-портов.
- Для поддержки транкинга можно настроить протокол GVRP для управления VLAN-сетями динамического добавления в базу данных VLAN.

Конфигурирование протокола GVRP

Протокол GVRP поддерживается только на COS коммутаторах. Он работает лишь на магистральных портах 802.1Q и используется главным образом для отсекания от VLAN-сетей трафика, который не нужно передавать между коммутаторами, соединенными магистральными каналами. Для настройки протокола GVRP используется описанная ниже последовательность действий.

1. Глобальное включение протокола GVRP.

Система COS	`set gvrp enable`

При стандартных установках протокол GVRP для коммутатора отключен. Прежде чем конфигурировать 802.1Q порты для GVRP-операций, необходимо включить этот протокол.

2. Конфигурирование порта для 802.1Q-операций.

Система COS	`set trunk` *mod/port* `[auto	desirable	on] [dot1q]`

Протокол GVRP работает только на портах, сконфигурированных для 802.1Q транкинга. Более подробные сведения по транкингу приведены в разделе "6.3 транкинг".

3. Конфигурирование протокола GVRP на порту.

Система COS	`set port gvrp` *mod/port* `enable`

Эта команда включает протокол GVRP на отдельном магистральном 802.1Q-порту. Чтобы протокол функционировал корректно, его необходимо настроить на обеих сторонах магистрального канала.

4. Конфигурирование регистрационного режима порта (*необязательно*).

Система COS	`set gvrp registration [normal	fixed	forbidden]` *mod/port*

Стандартно GVRP-порты функционируют в обычном режиме регистрации (`normal`). Такие порты для отсекания VLAN-сетей, передающих данные через магистральный 802.1Q-канал, используют GVRP-сообщения о подключении от соседних коммутаторов. Если устройство на другой стороне неспособно отправлять GVRP-сообщения или отсекать какую-либо из VLAN-сетей нежелательно, используется фиксированный режим (`fixed`). Порты в фиксированном режиме осуществляют передачу для всех VLAN-сетей, существующих в базе данных

коммутатора. Порты в запрещенном (forbidden) режиме передают данные только для сети VLAN 1.

Конфигурирование протокола GVRP для динамического создания VLAN-сетей

Как и протокол VTP, GVRP в целях поддержки транкинга способен создавать VLAN-сети на коммутаторах динамически. Если функция динамического создания VLAN-сетей включена, то коммутатор при получении GVRP-сообщений об отсутствующих в его базе данных VLAN-сетях добавляет их в базу данных.

1. Включение функции динамического создания VLAN-сетей *(необязательно)*.

Система CatOS	`set gvrp dynamic-vlan-creation enable`

Динамическое создание VLAN-сетей конфигурируется последовательно от коммутатора к коммутатору. Протокол GVRP не выполняет синхронизацию между коммутаторами, но для прохождения трафика между магистральными каналами только добавляет VLAN-сеты в устройства, в которых функция динамического создания VLAN-сетей включена. Для включения механизма динамического создания VLAN-сетей необходимо, чтобы все магистральные порты на коммутаторе имели тип 802.1Q, а также на них нужно включить протокол GVRP. Если в коммутаторе имеются какие-либо магистральные порты, не 802.1Q, или существующие 802.1Q-порты не настроены на протокол GVRP, эта функция не будет включена. VLAN-сети добавляются только для тех сообщений о подключении, которые получены через порт в обычном режиме регистрации. Кроме того, необходимо настроить протокол VTP в прозрачном режиме или отключить его, поскольку протокол VTP и механизм динамического создания VLAN-сетей не могут функционировать одновременно.

Внимание!

При включении функции динамического создания VLAN-сетей (`dynamic-vlan-creation`) магистральные порты 15/1 и 16/1 на коммутаторах серий 5000 и 6000 не рассматриваются как ISL-порты и не препятствуют работе функции.

Проверка GVRP-операций

После конфигурирования протокола GVRP для проверки его функционирования используется следующая команда:

Система CatOS	`show gvrp configuration`

Пример конфигурирования функции

В этом примере коммутатор `Access_1` подключен к коммутатору `Distribution_1` через 802.1Q-канал (рис. 6.4). Коммутатор `Distribution_1` также подключен к коммутатору `Core_1` через 802.1Q-канал. Протокол GVRP включается как на коммутаторе распределения, так и на основном коммутаторе, а также на всех GVRP-портах. Динамическое создание VLAN-сетей также включается на коммутаторах, и порт от коммутатора

Distribution_1 к Access_1 установлен в фиксированный GVRP-режим, поскольку устройство Access_1 не будет отправлять сообщения о подключении, а коммутатор распределения отсекал бы все VLAN-сети, если бы работал в обычном стандартном режиме.

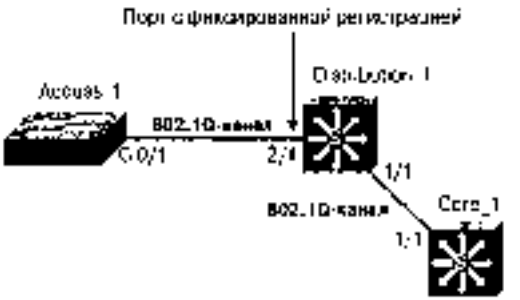


Рис. 6.4. Сетевая топология для конфигурации коммутаторов Access_1, Distribution_1 и Core_1

Пример конфигурации Catalyst OS для коммутатора Core_1

```
Core_1 (enable)>set vtp mode transparent
Core_1 (enable)>set trunk 1/1 on dot1q
Core_1 (enable)>set gvrp enable
Core_1 (enable)>set port gvrp 1/1 enable
Core_1 (enable)>set gvrp dynamic-vlan-creation enable
```

Пример конфигурации Catalyst OS для коммутатора Distribution_1.

```
Distribution_1 (enable)>set vtp mode transparent
Distribution_1 (enable)>set trunk 1/1 on dot1q
Distribution_1 (enable)>set trunk 2/1 on dot1q
Distribution_1 (enable)>set gvrp enable
Distribution_1 (enable)>set gvrp enable 1/1
Distribution_1 (enable)>set gvrp enable 2/1
Distribution_1 (enable)>set gvrp registration fixed 2/1
Distribution_1 (enable)>set gvrp dynamic-vlan-creation enable
```

Пример конфигурации системы IOS второго уровня для коммутатора Access_1.

```
Access_1 #config t
Access_1 (config)#interface gigabitethernet 0/1
Access_1 (config-if)#switchport mode trunk
Access_1 (config-if)#switchport trunk encapsulation dot1q
Access_1 (config-if)#end
Access_1#copy running-config startup-config
```

6.6: частные сети VLAN

- Частные сети VLAN позволяют обеспечить дополнительную защиту между устройствами в общей подсети
- Граничные частные сети VLAN (edge private VLAN) можно настроить для предотвращения соединений между устройствами на коммутаторах доступа.
- Частные сети VLAN можно настроить на устройствах Catalyst серий 6000 и 4000.

- Внутри частной VLAN-сети можно изолировать устройства для предотвращения соединения между устройствами внутри изолированной VLAN-сети.
- Внутри частной VLAN-сети можно создавать сообщества, допускающие соединения между некоторыми устройствами и предотвращающие их соединения с остальными.
- Чтобы разрешить связь с VLAN-сетями, расположенными за пределами сети, частные VLAN-сети связятся в соответствии неизбирательные порты (promiscuous ports).

Конфигурирование частных VLAN-сетей

Частные VLAN-сети обеспечивают механизм, позволяющий определять, какие устройства могут обмениваться данными внутри отдельной подсети. В частных VLAN-сетях для управления информационным обменом устройств используются вторичные VLAN-сети (secondary VLAN) `isolated` (изолированные) и `community` (VLAN-сообщество). Вторичные VLAN-сети являются первичной VLAN-сети (primary VLAN), а порты назначаются вторичным VLAN-сетям. Порты в изолированной VLAN-сети не могут обмениваться данными с каким-либо устройством во VLAN-сети, кроме неизбирательного порта. Порты, сконфигурированные во VLAN сообщества, способны обмениваться данными с другими портами в том же сообществе и неизбирательным портом. Порты, находящиеся в различных сообществах, не могут обмениваться данными друг с другом. При конфигурировании частных VLAN сетей выполняется определенная ниже последовательность действий:

1. Установка прозрачного режима протокола VTP.

Система COS	`set vtp mode transparent`

Система IOS	`vlan database`
	(режим привилегированного пользователя)
	`vtp transparent`
	(режим конфигурирования базы VLAN)

Прежде чем можно будет создать частную VLAN-сеть, необходимо настроить протокол VTP в прозрачном режиме. Частная VLAN-сеть конфигурируется в контексте отдельного коммутатора, и ее узлы не могут располагаться на других коммутаторах. Кроме того, частные VLAN-сети транспортируют сообщения в формате TLV, которые известны не всем типам коммутаторов Cisco.

2. Создание первичной частной сети.

Система COS	`set vlan` *primary_number* `pvlan-type primary`
Система IOS	`vlan` *primary_number* (режим глобальной конфигурации) `private-vlan primary` (режим конфигурирования сетей VLAN)

Прежде всего необходимо создать первичную частную VLAN-сеть. Номер первичной сети используется при последующих этапах для привязки вторичных VLAN-сетей к неизбирательным портам.

3. Создание изолированных VLAN-сетей и VLAN-сообществ.

Система COS	`set vlan` *secondary_number* `pvlan-type {isolated community | twoway-community}`
Система IOS	`vlan` *secondary_number* (режим глобальной конфигурации) `private-vlan {isolated | community}` (режим конфигурирования сетей VLAN)

Необходимо настроить изолированные VLAN-сети или VLAN-сообщества для назначения портов и управления трафиком. Вторичные номера всех таких VLAN-сетей должны быть уникальными, а также отличаться от первичного номера. Члены изолированной VLAN-сети могут обмениваться данными только с неизбирательным портом (или портами), преобразованным на этапе 6, тогда как члены сообщества VLAN могут обмениваться информацией с членами того же сообщества и с неизбирательными портами. Двустороннее сообщество (two-way community) функционирует подобно обычному, но обладает дополнительным свойством, разрешая спискам управления доступом проверять трафик, следующий в данную VLAN-сеть и из нее (для тампонирования), а также обеспечивает улучшенную защиту внутри частной VLAN-сети.

4. Привязка изолированных VLAN-сетей и VLAN-сообществ к первичной VLAN-сети

Система COS	`set pvlan` *primary_number secondary_number*
Система IOS	`vlan` *primary_number* (режим глобальной конфигурации) `private-vlan association` *secondary_number_list* `{add` *secondary_number_list*`}` (режим конфигурирования сетей VLAN)

Эта команда ставит в соответствие или привязывает вторичные VLAN-сети к первичной сети VLAN. В операционной системе IOS параметр `add` позволяет в дальнейшем привязывать другие сети.

5. Размещение портов в изолированных VLAN-сетях или VLAN-сообществах

Система COS	`set pvlan` *primary_number secondary_number mod/port {set}*

Система IOS	`interface` *type mod/port*
	(режим глобальной конфигурации)
	`switchport`
	`switchport mode private-vlan host`
	`switchport mode private-vlan host-association`
	primary_number secondary_number
	(три последние команды вводятся в режиме конфигурирования интерфейса)

После того как первичные и вторичные VLAN-сети созданы и связаны, необходимо назначить порты данной VLAN-сети. Для COS-коммутаторов также можно добавить интерфейс `sc0` к частной VLAN-сети.

6. Сопоставление изолированной VLAN-сети и VLAN-сообщества с неизбирательным портом (или портами)

Система COS	`set pvlan mapping` *primary_number secondary_number mod/port*

Система IOS	`interface` *type mod/port*
	(режим глобальной конфигурации)
	`switchport`
	`switchport mode private-vlan promiscuous`
	`switchport mode private-vlan mapping` *primary_number secondary_number*
	(три последние команды вводятся в режиме конфигурирования интерфейса)

После того как вторичным VLAN-сетям назначены порты, необходимо связать эти VLAN-сети с неизбирательным портом для доступа за пределы изолированной VLAN сети или VLAN-сообщества.

7. Связывание изолированной VLAN-сети и VLAN-сообщества с интерфейсом *функциональной платы многоуровневого коммутатора (Multilayer Switch Feature Card — MSFC)* (*необязательно*).

Система COS	`set pvlan mapping` *primary_number secondary_number 15/1* `session 15`
Система IOS	`private-vlan mapping` *primary_number secondary_number* (режим конфигурирования интерфейса)

Если в коммутаторе есть плата MSFC, то можно сопоставить с ней частные VLAN-сети. В коммутаторе, использующем COS, VLAN-сеть привязывается к порту 15/1 (или 16/1 для MSFC-платы в гнезде 2), а затем IP-адрес настраивается на VLAN-интерфейсе с номером первичной VLAN-сети. В IOS-коммутаторе необходимо перейти к VLAN-интерфейсу с номером первичной сети, а затем сопоставить с этим портом первичную и вторичные VLAN-сети.

Конфигурирование первичных граничных VLAN-сетей

Чтобы разрешить управление трафиком на коммутаторе, в устройстве 3500XL используется понятие защищенного порта (protected port). Защищенный порт коммутатора 3500XL не передает трафик другому защищенному порту того же коммутатора. Такое поведение подобно работе изолированной VLAN-сети, поскольку защищенные порты также не могут обмениваться данными друг с другом. Для настройки защищенного порта используются приведенные ниже команды.

1. Конфигурирование защищенного порта (*необязательно*)

   Для конфигурирования частной граничной VLAN-сети необходимо выбрать интерфейс и ввести команду `port protected`. Для проверки того, функционирует ли какой-либо порт в защищенном режиме, используется команда `show port protected`.

Система COS	Нет

Система IOS	`interface type mod/port`
	(режим глобальной конфигурации)
	`port protected`
	(режим конфигурирования интерфейса)

Проверка функционирования частной VLAN-сети

После конфигурирования частных VLAN-сетей для проверки их функционирования используются следующие команды:

Система COS	`show pvlan number`
	`show pvlan mapping`
	`show pvlan capability mod/port`

Система IOS	`show vlan private-vlan [type]`
	`show interface private-vlan mapping`
	`show interface type mod/port switchport`

Внимание!

К частным VLAN-сетям применим ряд правил и ограничений. Полный их перечень доступен по адресу www.cisco.com/univercd/cc/td/doc/product/lan/cat6000/sw_7_2/config_gd/vlans.htm#xtocid21.

Пример конфигурирования функции

На рис. 6.5 представлена сетевая диаграмма для рабочего примера конфигурации частной VLAN сети. В этом примере коммутатор Access_1 конфигурируется с портами 1 и 2 в защищенном режиме в сети VLAN 10. Сервер сети VLAN 10, подключенный к коммутатору Distribution_1, также находится в этой сети, что позволяет PC-станциям подключаться к серверу, но не друг к другу. Кроме того, на коммутаторе распределения частная сеть VLAN 901 создана с сообществом VLAN 901 и изолированной сетью VLAN 900. Сервер 2, подключенный к порту 3/46, и сервер 3 (порт 3/48) располагаются в сообществе VLAN, а серверы, подключенные к портам 3/1 и 3/2, должны размещаться в изолированной VLAN-сети. Все указанные устройства привязаны к маршрутизатору, подключенному к порту 1/2 и MSFC-порту 15/1 для интерфейса VLAN 90.

Пример конфигурации Catalyst OS для коммутатора Distribution_1.

```
Distribution_1 (enable)set vtp mode transparent
Distribution_1 (enable)set vlan 90 pvlan-type primary
Distribution_1 (enable)set vlan 900 pvlan-type isolated
Distribution_1 (enable)set vlan 901 pvlan-type community
Distribution_1 (enable)set pvlan 90 900
Distribution_1 (enable)set pvlan 90 901
Distribution_1 (enable)set pvlan 90 900 3/1-2
Distribution_1 (enable)set pvlan 90 901 3/46,3/48
Distribution_1 (enable)set pvlan mapping 90 900 1/2,15/1
Distribution_1 (enable)set pvlan mapping 90 901 1/2,15/1
Distribution_1 (enable)session 15
```

```
MSFC_Distr#enable
MSFC_Distr#config t
MSFC_Distr(config)#interface vlan 90
MSFC_Distr(config-if)#ip address 10.10.90.1 255.255.255.0
MSFC_Distr(config-if)#no shut
MSFC_Distr(config-if)#end
MSFC_Distr#copy running-config startup-config
```

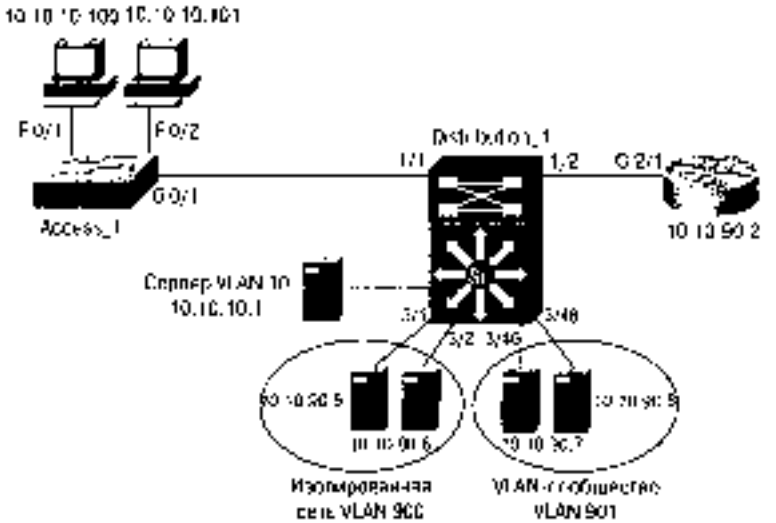


Рис. 6.3. Сетевая диаграмма для конфигурации частной VLAN-сети

Пример конфигурации Supervisor IOS для коммутатора Distribution_1.

```
Distribution_1#vlan database
Distribution_1(vlan)#vtp transparent
Distribution_1(vlan)#exit
Distribution_1#conf t
Distribution_1(config)#vlan 90
Distribution_1(config-vlan)#private-vlan primary
Distribution_1(config-vlan)#vlan 900
Distribution_1(config-vlan)#private-vlan isolated
Distribution_1(config-vlan)#vlan 901
Distribution_1(config-vlan)#private-vlan community
Distribution_1(config-vlan)#vlan 90
Distribution_1(config-vlan)#private-vlan association 900,901
Distribution_1(config-vlan)#interface range fastethernet 3/1 -2
Distribution_1(config-if)#switchport
Distribution_1(config-if)#switchport mode private-vlan host
Distribution_1(config-if)#switchport mode private-vlan host-
association 90 900
Distribution_1(config-if)#no shut
Distribution_1(config-if)#interface range fastethernet 3/46 ,3/48
Distribution_1(config-if)#switchport
Distribution_1(config-if)#switchport mode private-vlan host
Distribution_1(config-if)#switchport mode private-vlan host-
```

```
association 90 901
Distribution_1(config-if)#no shut
Distribution_1(config-if)#interface gigabitethernet 1/2
Distribution_1(config-if)#switchport
Distribution_1(config-if)#switchport mode private-vlan promiscuous
Distribution_1(config-if)#switchport mode private-vlan mapping 90
900,901
Distribution_1(config-if)#no shut
Distribution_1(config vif)#interface vlan 90
Distribution_1(config-if)#ip address 10.10.90.1 255.255.255.0
Distribution_1(config-if)#private-vlan mapping 90 900,901
Distribution_1(config-if)#no shut
Distribution_1(config-if)#end
Distribution_1 #copy running-config startup-config
```

Пример конфигурации IOS второго уровня для коммутатора Access_1

```
Access_1 #config t
Access_1 (config)#interface fastethernet 0/1
Access_1 (config-if)#switchport access vlan 10
Access_1 (config-if)#port protected
Access_1 (config)#interface fastethernet 0/2
Access_1 (config-if)#switchport access vlan 10
Access_1 (config-if)#port protected
Access_1 (config)#interface gigabitethernet 0/1
Access_1 (config-if)#switchport mode trunk
Access_1 (config-if)#switchport trunk encapsulation dot1Q
Access_1 (config-if)#end
Access_1#copy running-config startup-config
```

Дополнительная литература

Рекомендуемые ниже источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Кеннеди Кларк, Кевин Гамильтон, *Принципы коммутации в локальных сетях Cisco*. ИД "Вильямс", 2003.

Karen Webb. *Building Cisco Multilayer Switched Networks*, Cisco Press.

Tom Boyles and David Hucaby. *CCNP Switching Exam Certification Guide*, Cisco Press.

Обеспечение сетевой безопасности с помощью частных VLAN-сетей и списков контроля доступа VLAN (Securing Networks with Private VLANs and VLAN Access Control Lists): www.cisco.com/warp/public/473/90.shtml.

Стандарт GVRP (802.1Q): http://standards.ieee.org/reading/ieee/std/lanman/802.1Q-1998.pdf.

В этой главе...

- **7.1: принцип действия протокола STP.** В разделе разъясняется алгоритм распределенного связующего дерева в отношении выполняемых коммутатором процессов и решений.
- **7.2: конфигурирование протокола STP.** В этом разделе представлены основные этапы конфигурирования *протокола распределенного связующего дерева (Spanning Tree Protocol — STP).*
- **7.3: точная настройка конвергенции распределенного связующего дерева.** В этом разделе описаны более сложные этапы конфигурирования и тонкой настройки STP-сходимости.
- **7.4: навигация по топологии распределенного связующего дерева.** В разделе даны рекомендации, касающиеся определения местоположения корневого узла топологии распределенного связующего дерева, а также описания активной топологии вручную.

Глава 7

Протокол распределенного связующего дерева (STP)

7.1: принцип действия протокола STP

- Протокол STP обнаруживает и предотвращает формирование мостовых петель второго уровня. Параллельные маршруты могут существовать, но передача фреймов допускается только по одному из них.
- Протокол STP основан на стандарте мостового протокола IEEE 802.1D.
- Коммутаторы запускают по одному экземпляру STP на каждую VLAN-сеть с помощью алгоритма PVST (*Per-VLAN Spanning Tree — отдельные экземпляры распределенного связующего дерева для разных сетей VLAN*). PVST-алгоритм между коммутаторами требует использования ISL-транкинга.
- В магистральных каналах IEEE 802.1Q разрешено использование только одного экземпляра STP для всех сетей. *Общее распределенное связующее дерево (Common Spanning Tree — CST)* связывается посредством сети VLAN 1.
- Функция PVST+ является частным расширением, созданным корпорацией Cisco, которое позволяет коммутаторам взаимодействовать между CST и PVST. *Блоки данных мостового протокола (Bridge Protocol Data Units — BPDU)* режима PVST отправляются по туннелю через магистральный 802.1Q-канал. Коммутаторы Catalyst стандартно используют режим PVST+.
- *Многоэкземплярный протокол распределенного связующего дерева (Multiple Instance Spanning Tree Protocol — MISTP)* также является частным протоколом корпорации Cisco, который допускает использование одного экземпляра STP для одной или нескольких VLAN-сетей посредством функции отображения (mapping). Такой подход позволяет ускорить конвергенцию с меньшей нагрузкой на процессор и меньшим количеством блоков BPDU. Протокол MISTP отбрасывает BPDU-блоки PVST+.
- *MISTP-PVST+* — гибридный STP-режим, который используется для перехода между режимами PVST+ и MISTP в сети. BPDU-блоки обоих режимов распознаются и не уничтожаются.

- *Многостековые распределенные связующие деревья (Multiple Spanning Tree – MST)*, основанные на стандарте IEEE 802.1s, расширяют *ускоренный протокол распределенного связующего дерева (802.1w Rapid Spanning Tree Protocol — RSTP)* до нескольких экземпляров STP.
  - Режим MST обладает обратной совместимостью с STP-режимами 802.1D, 802.1w и PVST+.
  - Коммутаторы, сконфигурированные с общими VLAN-назначениями и экземпляром STP, формируют отдельную *MST-область (region)*.
  - MST-структуры для обеспечения взаимодействия способны генерировать блоки BPDU PVST+.
  - Режим MST поддерживает до 16 экземпляров STP.
- Коммутаторы отправляют BPDU-блоки через все порты по одному разу в течение каждого интервала hello-таймера (стандартно — 2 секунды).
- Блоки BPDU не перенаправляются коммутатором, они применяются только для дальнейшего вычисления и генерирования BPDU.
- Коммутаторы отправляют два типа BPDU-блоков:
  - конфигурационные сообщения BPDU,
  - *уведомления об изменении топологии (Topology Change Notification — TCN BPDU)*.

Внимание!

BPDU-блоки отправляются на фиксированный мультикастовый STP-адрес 01-80-c2-00-00-00, используя уникальный MAC-адрес каждого порта коммутатора в качестве адреса отправителя.

STP-процесс

1. **Выбор корневого моста (root bridge)**. Коммутатор с наименьшим идентификатором моста становится корневым узлом распределенного связующего дерева. *Идентификатор моста (bridge ID)* составляется из двухбайтового значения приоритета и шестибайтового MAC-адреса. Приоритет может варьироваться от 0 до 65535, стандартное значение – 32768.
2. **Выбор корневого порта** (root port). Каждый некорневой коммутатор путем определения порта с наименьшей стоимостью корневого маршрута выбирает корневой порт или порт, "ближайший" к корневому мосту. Значение стоимости транспортируется в составе блока BPDU. Каждый некорневой коммутатор маршрута добавляет стоимость своего локального порта, на котором принимается BPDU-блок. Стоимость корневого маршрута накапливается по мере создания новых BPDU-блоков.
3. **Выбор назначенного порта (designated port)**. Один порт коммутатора в каждом сетевом сегменте выбирается для обработки трафика данного сегмента. Порт, объявивший наименьшую стоимость корневого маршрута в сегменте, становится назначенным.
4. **Удаление мостовых петель**. Порты коммутатора, не являющиеся ни корневыми, ни назначенными, переводятся в состояние блокировки. На этом этапе уничтожаются все возможные мостовые петли.

Схема разрешения конфликтов в STP

Если какое-либо STP-решение имеет идентичные или совпадающие состояния, то окончательное решение базируется на описанной ниже последовательности условий.

1. Выбор наименьшего BID-идентификатора.
2. Выбор наименьшей стоимости корневого маршрута.
3. Выбор наименьшего BID-идентификатора отправителя.
4. Выбор наименьшего идентификатора порта.

Значения стоимости маршрутов

Стандартно порты коммутаторов обладают определенными значениями стоимости маршрутов, которые указаны в табл. 7.1.

Таблица 7.1. Стоимость маршрутов для портов коммутатора

Скорость порта	Стандартная стоимость порта в "сокращенном режиме"	Стандартная стоимость порта в "расширенном режиме"
4 Мбит/с	250	Нет
10 Мбит/с	100	2 000 000
16 Мбит/с	62	Нет
45 Мбит/с	39	Нет
100 Мбит/с	19	200 000
155 Мбит/с	14	Нет
622 Мбит/с	6	Нет
1 Гбит/с	4	20 000
10 Гбит/с	2	2000
100 Гбит/с	Нет	200
1000 Гбит/с (1 Тбит/с)	Нет	20
10 Тбит/с	Нет	2

Стандартно коммутаторы Catalyst в режиме PVST+ используют "сокращенный режим" или 16-битовые значения стоимости маршрута или порта. Шкала сокращенного режима достаточна в случае, если скорости портов в сети меньше 1 Гбит/с. Однако, если имеются какие-либо порты, скорость которых равна или выше 10 Гбит/с, следует настроить все коммутаторы в сети на использование "расширенного режима" или 32-битовой шкалы стоимости маршрута. В таком случае будет обеспечено согласованное вычисление стоимости корневого маршрута на всех коммутаторах. Коммутаторы в режимах MISTP, MISTP-PVST+ и MST автоматически используют значения расширенного режима.

Внимание!

В IEEE-стандарте применяется нелинейная шкала отношения полосы пропускания порта для отдельного канала к стоимости его порта. В протоколе STP объединенные каналы, такие как Fast EtherChannel и Gigabit EtherChannel, рассматриваются как отдельный ка-

ней с агрегированной полосой пропускания индивидуальных каналов, поэтому необходимо помнить, что стоимость порта или маршрута для объединенного EtherChannel-канала основывается на суммарной полосе пропускания. Например, двухканальный Fast EtherChannel обладает полосой пропускания 200 Мбит/с и стоимостью маршрута, равной 12. Четырехканальный Gigabit EtherChannel обладает полосой пропускания 4 Гбит/с и стоимостью маршрута, равной двум. Отношения агрегированных полос пропускания, таких, как EtherChannel-каналов, и стоимости портов к значениям отдельных или индивидуальных каналов приведены в табл. 7.1.

Состояния портов в STP

Каждый порт коммутатора последовательно проходит ряд состояний.

1. **Отключен.** Административно отключенные порты или порты, отключенные ввиду возникновения сбоя. В режиме MST такое состояние называется *отбрасывающим (discarding)*.
2. **Блокировка.** Состояние, которое применяется после инициализации порта. Порт в состоянии блокировки не может принимать или передавать данные, добавлять MAC-адреса в свою адресную таблицу, он может лишь принимать блоки BPDU. При обнаружении мостовой петли либо потери посылок статуса корневого или назначенного порт возвращается в состояние блокировки. В режиме MST такое состояние называется *отбрасывающим*.
3. **Прослушивание.** Если порт может стать корневым или назначенным, он переводится в состояние прослушивания, при котором не может принимать или передавать данные, добавлять MAC-адреса в свою адресную таблицу, но может получать и отправлять BPDU-блоки. В режиме MST такое состояние называется *отбрасывающим*.
4. **Состояние самообучения.** По истечении таймера задержки передачи (стандартно – 15 секунд) порт входит в состояние самообучения (learning state). Он не может передавать данные, но может получать и отправлять BPDU-блоки. В этом состоянии порт может изучать MAC-адреса и добавлять их в адресную таблицу.
5. **Состояние передачи.** По истечении следующей задержки передачи (стандартно – 15 секунд) порт переходит в состояние передачи, при котором он может отправлять и принимать данные, изучать MAC-адреса, а также отправлять и принимать BPDU-блоки.

Изменения STP-топологии

- Если порт переводится в состояние передачи (кроме случая, когда включена функция PortFast), отправляется уведомление об изменении топологии.
- Если порт переводится из состояния самообучения или передачи в состояние блокировки, отправляется уведомление об изменении топологии.
- Для объявления об изменении топологии коммутатор периодически (период равен hello-интервалу) отправляет в свой корневой порт TCN BPDU-блоки. Отправка BPDU происходит до тех пор, пока не будет получено TCN-подтверждение от соседа вышестоящего выделенного моста. Соседи продолжают ретранслировать TCN BPDU-блок на свои корневые порты до тех пор, пока он не будет получен корневым мостом.

- Корневой мост информирует все распределенное связующее дерево об изменении топологии путем отправки конфигурационного BPDU-блока с установленным битом *изменения топологии (Topology Change — TC)*. В результате все нижестоящие коммутаторы сокращают таймеры старения адресных таблиц на длительность задержки передачи (15 секунд) от стандартного значения (300 секунд). Неактивные MAC-адреса в таком случае удаляются из таблиц быстрее, чем обычно.

Усиление стабильности протокола STP

- Служба STP Root Guard (служба защиты корневого моста) может быть полезна при выборе местоположения корневого моста и при подтверждении его уникальности в коммутируемой сети. Когда данная функция включена на каком-либо порту, при получении лучшего BPDU-блока порт отключается, что препятствует другим коммутаторам незапланированно становиться корневыми.
- Служба STP Root Guard должна быть включена на всех портах, где не следует обнаруживать корневой мост. Это позволяет сохранить текущий выбор основного и дополнительного корневых мостов.
- *Обнаружение однонаправленной передачи в канале (Unidirectional Link Detection — UDLD)* — способ обнаружения канала, в котором передача осуществляется только в одном направлении, что позволяет предотвратить возникновение мостовых петель и "черных дыр" для трафика, которые обычно не обнаруживаются и не предотвращаются протоколом STP.
- Механизм UDLD функционирует на втором уровне путем отправки пакетов, содержащих идентификаторы устройства и порта, соседям, подключенным к портам коммутатора. Также любые UDLD-пакеты, получаемые от соседнего устройства, отражаются для того, чтобы данное устройство получило подтверждение о том, что оно опознано. UDLD-сообщения отправляются в течение *интервала сообщений (message interval)*, стандартная длительность которых обычно равна 15 секундам.
- Механизм UDLD функционирует в двух режимах:
  - **Обычный режим.** Однонаправленные каналы обнаруживаются и объявляются ошибочными, но никакие действия не предпринимаются.
  - **Агрессивный режим.** Однонаправленные каналы обнаруживаются, объявляются ошибочными и отключаются после восьми попыток (по одной в секунду в течение восьми секунд) переустановки канала. Отключенные порты необходимо включать вручную.
- Функция STP Loop Guard обнаруживает отсутствие BPDU-блоков на корневом и альтернативном корневом портах. Назначенные порты временно отключаются, что препятствует их переходу в назначенные порты и состояние передачи.
- Функцию STP Loop Guard необходимо включить на корневом и альтернативном корневом портах (оба неназначенные) для всех возможных вариантов активной STP-топологии.

Пример функционирования протокола STP

В качестве примера функционирования протокола STP рассмотрим сеть, состоящую из трех коммутаторов Catalyst, подключенных по схеме треугольника (рис. 7.1).

Корневые порты отмечаются литерами RP, назначенные порты — DP, метки F соответствуют портам в состоянии передачи, а X — портам, которые находятся в состоянии блокировки.

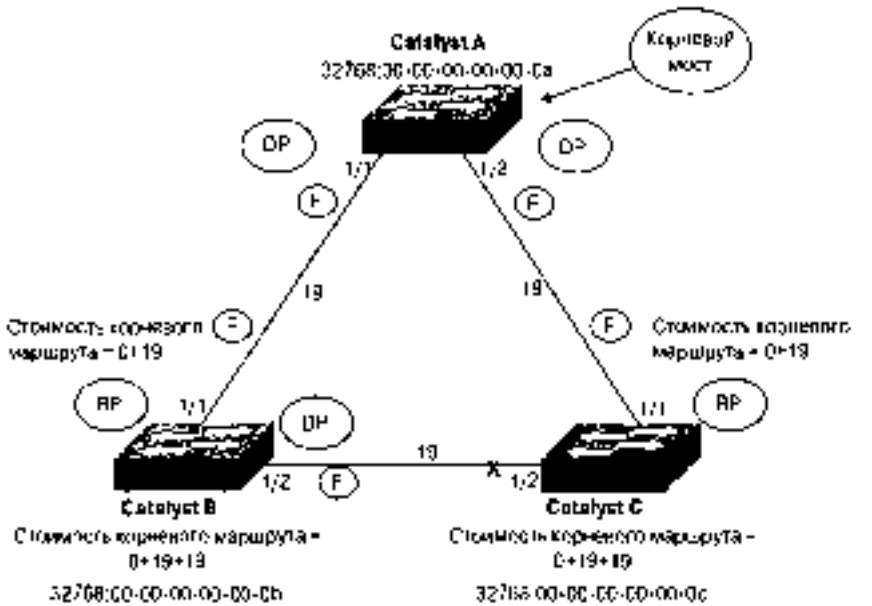


Рис. 7.1. Структура сети для примера функционирования протокола STP

Рассмотрим выполнение алгоритма распределенного связующего дерева.

1. **Выбор корневого моста.** Все три коммутатора имеют равные значения приоритета моста (стандартный приоритет равен 32768). В то же время коммутатор Catalyst A имеет наименьший MAC-адрес (00-00-00-00-00-0a), поэтому он становится корневым мостом.
2. **Выбор корневых портов.** На каждом коммутаторе вычисляются стоимости корневых маршрутов. На коммутаторе Catalyst B порт 1/1 имеет стоимость корневого маршрута 0+19. Порт 1/1 коммутатора Catalyst C также имеет стоимость корневого маршрута, равную 0+19.
3. **Выбор назначенных портов.** По определению все порты корневого моста становятся назначенными портами для своих сегментов. Следовательно, порты 1/1 и 1/2 коммутатора Catalyst A являются назначенными. Порт 1/2 коммутатора Catalyst B и порт 1/2 коммутатора Catalyst C совместно используют один сегмент. Требуется, чтобы один из указанных портов стал назначенным. Стоимость корневого маршрута для каждого из портов равна 0+19+19, или 38, что делает невозможным выбор одного из них. Наименьший отправленный идентификатор моста позволяет решить эту проблему, и порт 1/2 коммутатора Catalyst B (имеющего наименьший адрес из двух) становится назначенным.
4. **Все порты, не являющиеся ни корневыми, ни назначенными, переходят в состояние блокировки.** Единственный порт, который не является ни корневым, ни назначенным, — это порт 1/2 коммутатора Catalyst C. Порт переходит в состояние блокировки (отмеченное на схеме литерой X).

7.2: конфигурирование протокола STP

1. Включение или отключение протокола STP (*необязательно*).

Система COS	`set spantree (enable | disable) [vlan]`
Система IOS	`[no] spanning-tree [vlan vlan]` (режим глобальной конфигурации)

Протокол STP стандартно включен в сети VLAN 1 и в любых недавно созданных VLAN-сетях. Если сеть VLAN не указана, то протокол STP включается или отключается во всех VLAN сетях. Следует помнить о том, что при отключенном механизме STP жесткие петли не обнаруживаются, а их возникновение не предотвращается. Протокол STP следует включать всегда.

2. Установка STP-режима для коммутатора (только для коммутаторов Catalyst серий 4000 и 6000) (*необязательно*).

Система COS	`set spantree mode {mistp | pvst+ | mistp-pvst+ mst}`
Система IOS	Нет

При стандартных настройках для одного экземпляра STP в каждой VLAN-сети все коммутаторы Catalyst выполняют протокол STP в режиме PVST+. Для настройки других STP-режимов используются ключевые слова `mistp` (MISTP), `mistp-pvst+` (взаимодействие MISTP-PVST+) и `mst` (MST).

3. Активизация MST-экземпляра (только коммутаторы Catalyst 4000 и 6000) (*только MST-режим*).

а) Идентификация MST-области.

Система COS	`set spantree mst config {name name} {revision number}`
Система IOS	Нет

MST-область идентифицируется по имени (`name`) (текстовая строка длиной до 32 символов). Если имя не задано, то имя области не используется. Чтобы указать количество изменений конфигурации области, можно использовать номер ревизии области (region revision number). Номер ревизии (`number`) (от 0 до 65535, стандартно 1) указывается явно и не увеличивается автоматически при изменениях области.

б) Назначение одной или нескольких VLAN-сетей экземпляру.

Система COS	`set spantree mst instance vlan vlan`
Система IOS	Нет

Номер VLAN сети (`vlan`) (от 1 до 1005, от 1025 до 4094) сопоставляется с MST экземпляром (`instance`) (от 0 до 15). Это назначение сохраняется в буфере MST области до тех пор, пока не будут внесены изменения.

в) Фиксация назначения области.

Система COS	`set spantree mst config commit`
Система IOS	Нет

Конфигурационные изменения MST области помещаются в буфер редактора, который выделяется пользователю, осуществляющему эти изменения. Их необходимо зафиксировать до того, как они станут активными. При фиксации изменений также освобождается буфер редактора и появляется возможность инициировать новый сеанс редактирования.

3. Отмена последних изменений конфигурации области (*необязательно*).

Система COS	`set spantree mst config rollback [force]`
Система IOS	Нет

Если конфигурационные изменения для MST-области были сделаны ошибочно, их можно отменить при помощи ключевого слова `rollback`. Отмена возможна только для тех изменений, которые еще не были зафиксированы или введены в действие. В ситуации, когда какой-либо пользователь внес изменения и продолжает удерживать буфер редактора, можно при помощи ключевого слова `force` освободить буфер и удалить изменения.

4. Указание корневого моста (*необязательно*).

Внимание!

Корневой мост (и вторичные корневые мосты) следует размещать вблизи от "центра" сети, для того чтобы вычислять оптимальную топологию распределенного связующего дерева. Как правило, корневой мост располагается на основном уровне или на уровне распределения сети. Если не конфигурировать размещение корневого моста вручную, то корневым становится коммутатор с наименьшим BID-идентификатором. В таком случае почти всегда создается неэффективная топология распределенного связующего дерева.

Система COS	PVST+: `set spantree root [secondary] {vlans} [dia` *net-diameter*`] [hello` *hello-time*`]` MISTP: `set spantree root [secondary] mistp-instance` `instance [dia` *net-diameter*`] [hello` *hello-time*`]` MST: `set spantree root [secondary] mst` *instance* `[dia` *net-diameter*`] [hello` *hello-time*`]`
Система IOS	PVST+: `spanning-tree vlan` *vlan* `root {primary | secondary} [diameter` *net-diameter* `[hello-time` *hello-time*`]]` (режим глобальной конфигурации)

Коммутатор вынужден стать основным корневым мостом для VLAN-сетей (сети с номерами от 1 до 1005 и с 1025 по 4094) или для указанных STP-экземпляров (с 1 по 16) (если VLAN-сеть не указывается, используется сеть VLAN 1). Значение приоритета моста модифицируется следующим образом: если приоритет превышает 8192, то значение устанавливается равным 8192; если значение приоритета меньше 8192, оно устанавливается меньшим приоритета текущего корневого моста. Можно использовать ключевое слово `secondary` для размещения дополнительного или резервного корневого моста (тогда, когда основной корне-

этот мост выйдет из строя. В данном случае приоритет моста устанавливается равным 16384. (Для MST приоритет корневого моста устанавливается равным 24576, а приоритет вторичного корневого моста — равным 28672.)

Ключевое слово `dia` определяет диаметр либо максимальное количество мостов или коммутаторов между двумя конечными точками сети (от 1 до 7, стандартно 7). Также можно установить hello-интервал (стандартно 2 секунды). Установка диаметра сети приводит к тому, что другие STP-таймеры автоматически пересчитываются и изменяются. С помощью других команд можно явно отрегулировать таймеры, но настройка диаметра сети позволяет избежать сложности расчета таймеров.

Внимание!

Эта команда операционной системы Supervisor IOS недоступна на коммутаторах семейства Catalyst 2900XL и 3500XL.

5. Регулировка приоритета моста (*необязательно*)

Система COS	PVST+: **set spantree priority** *priority vlans* MISTP: **set spantree priority** *priority* **mistp-instance** *instance-list* MST: **set spantree priority** *priority* **mst** *instance-list*
Система IOS	PVST+: **spanning-tree vlan** *vlan* **priority** *priority* (режим глобальной конфигурации)

Кроме того, можно непосредственно модифицировать приоритет моста для достижения значений, отличных от автоматически определенных величин приоритета основного или дополнительного корневого моста. Приоритет можно устанавливать отдельно для каждой VLAN-сети и экземпляра. Экземпляры можно указывать в виде списка (*instance-list*) как один или несколько экземпляров, разделенных запятыми, или в виде диапазона номеров, заданного с помощью дефиса.

Чтобы перевести коммутатор в режим корневого, необходимо выбрать приоритет так, чтобы приоритет корневого моста был ниже приоритетов всех остальных коммутаторов в данной VLAN-сети или STP-экземпляре. Значения приоритета моста варьируются в диапазоне от 0 до 65535 (стандартный приоритет равен 32768) для PVST+, для MISTP-режима приоритет выбирается из списка значений: 0 (наивысший приоритет), 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344 и 61440 (наименьший приоритет).

6. Предотвращение перехода других коммутаторов в режим корневого моста STP (*необязательно*)

Система COS	**set spantree guard** {**root** | **none**} *mod/port*
Система IOS	**spanning-tree guard** {**root** | **none**} или **spanning-tree rootguard** (режим конфигурирования интерфейса)

Служба STP Root Guard будет включена на порту или интерфейсе. Если другой мост, подключенный к данному порту, попытается стать корневым, порт будет переведен в STP-состояние *root-inconsistent* (прослушивание). Если на пор-

ту более не обнаруживаются BPDU-блоки, он переводится обратно в нормальное состояние.

На коммутаторах семейства Catalyst 2900XL и 3500XL используется ключевое слово **rootguard**.

7. **Настройка стоимости корневого маршрута** (*необязательно*).

   а) Установка шкалы стоимости порта (только коммутаторы Catalyst 4000 и 6000) (*необязательно*)

Система COS	**set spantree defaultcostmode** {**short** | **long**}
Система IOS	**spanning-tree pathcost defaultcost-method** {**long** | **short**} (режим глобальной конфигурации)

   Стандартно коммутаторы в режиме PVST+ используют короткие, или сокращенные (**short**), 16-битовые значения стоимости порта. Если используются какие-либо порты с полосой пропускания 10 Гбит/с или более, то на каждом коммутаторе в сети следует установить шкалу стоимости порта для расширенных (**long**), 32-битовых значений. В MISTP, MISTP-PVST+ и MST стандартно используется расширенный режим.

   б) Установка стоимости порта для всех VLAN-сетей или экземпляров.

Система COS	**set spantree portcost** *mod/port cost* [**mst**]
Система IOS	**spanning-tree cost** *cost* (режим конфигурирования интерфейса)

   Стоимость порта может быть установлена в значение *cost* (от 1 до 65535 сокращенный, или MISTP-режим, от 1 до 2000000 — расширенный режим) для всех VLAN-сетей или STP-экземпляров. Ключевое слово **mst** обозначает порт, используемый в режиме MST.

   в) Установка стоимости порта в отдельных VLAN-сетях или экземплярах.

Система COS	PVST+: **set spantree portvlancost** *mod/port* [**cost** *cost*] [*vlan-list*] MISTP: **set spantree portinstancecost** *mod/port* [**cost** *cost*] [*instances*] MST: **set spantree portinstancecost** *mod/port* [**cost** *cost*] **mst** [*instances*]
Система IOS	PVST+: **spanning-tree vlan** *vlan-id* **cost** *cost* (режим конфигурирования интерфейса)

   Стоимость порта может быть установлена в значение *cost* (от 1 до 65535 — короткий режим, от 1 до 2000000 — длинный режим) для VLAN-сети *vlan-id*, списка VLAN-сетей *vlan-list* или STP-экземпляра (с 0 по 15).

8. **Точная настройка приоритета порта** (*необязательно*).

   а) Установка приоритета порта для всех VLAN-сетей или экземпляров

Система COS	`set spantree portpri` *mod/port priority* `[mst]`
Система IOS	`spanning-tree port-priority` *port-priority* (режим конфигурирования интерфейса)

Приоритет порта может устанавливаться в значение *priority* (от 0 до 63) — для COS или от 2 до 255 — для IOS). Чтобы указать, что порт используется для MST-режима, используется ключевое слово `mst`.

б) Установка приоритета порта для отдельных VLAN-сетей или экземпляров.

Система COS	PVST+: `set spantree portvlanpri` *mod/port priority [vlans]* MISTP: `set spantree portinstancepri` *mod/port priority [instances]* MST: `set spantree portinstancepri` *mod/port priority* `mst` *[instances]*
Система IOS	PVST+: `spanning-tree vlan` *vlan-list* `port-priority` *priority* (режим конфигурирования интерфейса)

Приоритет порта может устанавливаться в значение *priority* (от 0 до 63 — для COS или от 2 до 255 — для IOS) для VLAN-сети *vlan-id*, списка сетей *vlan-list* или для STP-экземпляра (с 0 по 15).

9. Активизация MISTP-экземпляра (*только для MISTP-режима*)

а) Включение MISTP-экземпляра.

Система COS	`set spantree enable mistp-instance {`*instance* `, all}`
Система IOS	Нет

MISTP-экземпляр 1 стандартно включен. Другие экземпляры можно включить, используя номер экземпляра (*instance* от 1 до 16) или ключевое слово `all`.

б) Назначение VLAN-сетей MISTP-экземпляру.

Система COS	`set vlan` *vlan-list* `mistp-instance {`*instance* `| none}`
Система IOS	Нет

Чтобы сопоставить одну или несколько VLAN-сетей с одним MISTP-экземпляром, можно задавать номера в виде списка *vlan-list*. Если случайно одна VLAN-сеть будет назначена нескольким экземплярам, то все ее порты будут переведены в состояние STP-блокировки. Чтобы отменить назначение VLAN-сетей каким-либо экземплярам, можно использовать ключевое слово `none`.

10. Обнаружение однонаправленных соединений с помощью функции UDLD (*необязательно*).

а) Включение функции UDLD на коммутаторе.

Система COS	`set udld {enable  disable}`
Система IOS	`udld {enable	aggressive}` (режим глобальной конфигурации)

Стандартно функция UDLD отключена. Прежде чем использовать ее на определенных портах, ее необходимо включить. В операционной системе Supervisor IOS допускается использование ключевого слова `aggressive` для глобального включения агрессивного режима UDLD на всех волоконно-оптических Ethernet-интерфейсах.

б) Регулировка интервала UDLD-сообщений (*необязательно*).

Система COS	`set udld interval` *interval*
Система IOS	`udld message time` *interval* (режим глобальной конфигурации)

Интервал UDLD-сообщений может быть установлен равным значению параметра *interval* (от 7 до 90 секунд, стандартные значения равны 15 и 60 секунд для систем COS и Supervisor IOS соответственно)

в) Включение функции UDLD на определенных портах.

Система COS	`set udld {enable | disable}` *mod/ports*
Система IOS	`udld {enable | disable}` (режим конфигурирования интерфейса)

После глобального включения на коммутаторе функции UDLD она стандартно включается также на всех волоконно-оптических Ethernet-портах. На всех Ethernet-портах для витой пары функция UDLD стандартно отключена.

г) Включение агрессивного режима UDLD на определенных портах (*необязательно*).

Система COS	`set udld aggressive-mode {enable | disable}` *mod/ports*
Система IOS	`udld aggressive` (режим конфигурирования интерфейса)

После включения на каком-либо порту агрессивного режима порт отключается, если обнаружено однонаправленное соединение. После устранения проблемы порт необходимо включить вручную. В операционной системе Supervisor IOS для включения всех портов, отключенных функцией UDLD, используется команда `udld reset` EXEC-режима.

11. Повышение стабильности протокола STP с помощью функции Loop Guard (Защита от петель) (*необязательно*).

Система COS	`set spantree guard loop` *mod/ports*
Система IOS	Нет

Службу Loop Guard следует включать только на тех портах, о которых точно известно, что они корневые или альтернативные корневые. Например, внешние порты коммутатора уровня доступа всегда будут корневыми или альтернативными корневыми, поскольку находятся наиболее близко к корневому мосту. (При этом предполагается, что корневой мост расположен вблизи центра сети.)

Отображение сведений о протоколе STP

В табл. 7.2 приведен список команд коммутаторов, которые можно использовать для отображения полезной информации о работе протокола STP.

Таблица 7.2. Команды коммутатора для отображения STP-информации

Функция отображения	Операционная система коммутатора	Команда
Протокол STP для определенной VLAN-сети	COS	`show spantree vlan active`
	IOS	`show spanning-tree vlan vlan`
STP-состояние для всех VLAN-сетей на магистральном канале	COS	`show spantree mod/num`
	IOS	`show spanning-tree interface mod/num`
STP-статистика для VLAN-сети на каком-либо порту	COS	`show spantree statistics mod/num vlan`
	IOS	Нет
Порты в состоянии блокировки	COS	`show spantree blockedports [vlan]`
	IOS	Нет
Журнал STP-событий	COS	`set logging level spantree severity`
	IOS	Нет

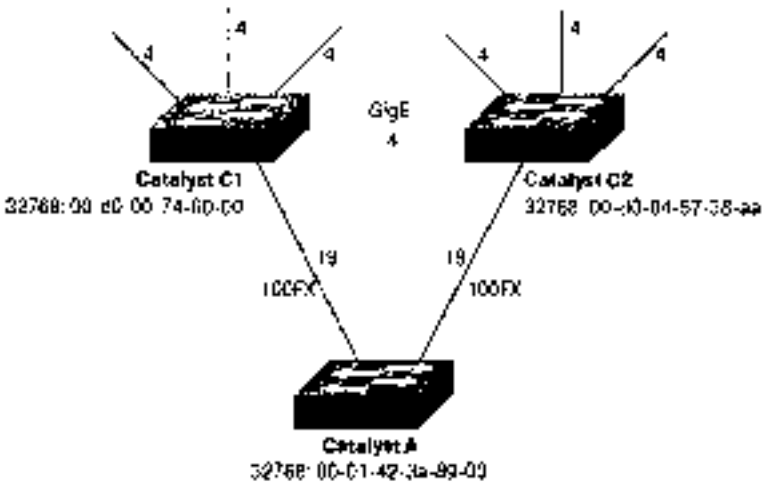
Примеры конфигурирования протокола STP

Считается хорошим правилом конфигурировать для конкретной VLAN-сети один коммутатор в качестве основного корневого моста, а другой — в качестве дополнительного. Предположим, при реализации сети эта рекомендация по какой-либо причине не соблюдена. Что может произойти, если коммутаторы самостоятельно организуют топологию распределенного связующего дерева на основании стандартных параметров протокола STP?

Неадекватное размещение корневого устройства распределенного связующего дерева

В верхней части рис. 7.2 показан пример сети, состоящей из трех коммутаторов Catalyst, подключенных по схеме треугольника. Коммутаторы Catalyst C1 и C2 формируют основной уровень сети, тогда как коммутатор Catalyst A подключается к конечным пользователям на уровне доступа. (Устройства C1 и C2 также могут рассматриваться как коммутаторы уровня распределения в том случае, если во всей территориальной сети нет четко ограниченного основного уровня. В любом случае их следует рассматривать как наивысший уровень или магистраль данной сети.)

Как и следовало ожидать, между коммутаторами основы и другими коммутаторами имеются каналы Gigabit Ethernet. В то же время в качестве внешних каналов от коммутатора Catalyst A в основу сети используются каналы Fast Ethernet.



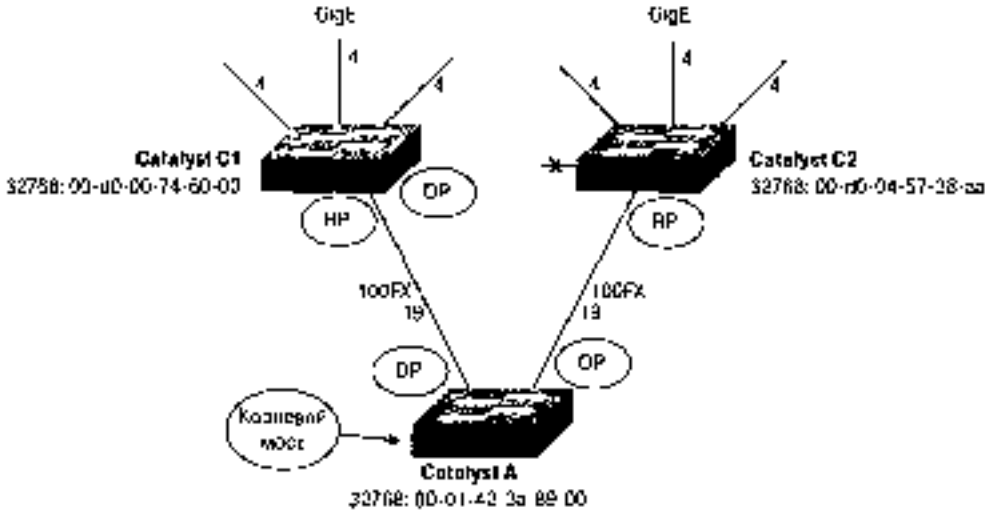


Рис. 7.2. Сетевая диаграмма, демонстрирующая неверное размещение корневого устройства распределенного связующего дерева

При выборе корневого моста коммутатор Catalyst A побеждает, поскольку имеет наименьший MAC-адрес. (Все коммутаторы имеют стандартное значение приоритета — 32768.) Оба внешних порта коммутатора A становятся назначенными, поскольку коммутатор в настоящее время является корневым. Нисходящие каналы (downlinks) от коммутаторов C1 и C2 к коммутатору A становятся корневыми портами. Коммутатор C1 определяет свой Gigabit Ethernet-канал к коммутатору C2 назначенным портом,

поскольку имеет меньший BID-идентификатор. Кроме того, коммутатор C2 принужден перевести свой Gigabit Ethernet-канал к коммутатору C1 в состояние блокировки, поскольку он не является ни корневым ни назначенным портом. Состояние каналов показано в нижней части схемы.

Очевидно, в данном случае создана неэффективная топология, поскольку весь трафик, проходящий через основу сети, вынужден пересекать низкоскоростные каналы коммутатора A. Кроме того, коммутатор A, будучи коммутатором уровня доступа, вероятно, обладает меньшей мощностью, чем коммутаторы основного уровня.

Для исправления описанной ситуации следует разместить корневой мост STP в пределах основы сети или наивысшего уровня сетевой иерархии. Например, для сети VLAN 10 на коммутаторе C1 это можно сделать с помощью приведенных ниже команд.

Система CatOS	`set spantree root 10`
Система IOS	`spanning-tree vlan 10 root primary` (режим глобальной конфигурации)

В качестве альтернативного решения можно явно задать приоритеты моста с помощью следующих команд (доступны на всех моделях коммутаторов Catalyst).

Система CatOS	`set spantree priority 8192 10`
Система IOS	`spanning-tree vlan 10 priority 8192` (режим глобальной конфигурации)

Балансировка нагрузки при помощи протокола распределенного связующего дерева

На рис. 7.3 представлена сетевая диаграмма, содержащая три коммутатора, подключенных по схеме треугольника. Каждый канал между коммутаторами является магистральным и поддерживает две VLAN-сети. Коммутаторы конфигурируются таким образом, что нагрузка двух VLAN-сетей распределяется по доступным магистральным каналам. В нижней части схемы показаны варианты результирующей топологии распределенного связующего дерева для сетей VLAN 100 и 101.

Коммутатор распределения D1 выбирается в качестве корневого моста. Некоторые пользователи, подключенные к коммутатору доступа Catalyst A1, находятся в сети VLAN 100, тогда как остальные — в сети VLAN 101. Нужно направить трафик сети VLAN 100 к коммутатору распределения Catalyst D1, а трафик VLAN 101 — к коммутатору Catalyst D2.

Внимание!

Коммутатор D1 выбран в качестве корневого моста для обеих сетей VLAN для простоты, а также для того, чтобы продемонстрировать использование регулировки стоимости порта при распределении нагрузки. Можно настроить коммутатор D1 в качестве корневого моста для сети VLAN 100, а D2 — для VLAN 101. Результирующая топология будет той же, но в таком случае нет необходимости регулировать стоимость портов в коммутаторе A1.

Дополнительное преимущество заключается в том, что два магистральных канала будут резервировать друг друга в случае отказа одного из них. Как только в одном из магистральных каналов возникает сбой, другой канал переходит из состояния блокировки в состояние передачи и транспортирует трафик для обеих сетей VLAN 100 и 101. Кроме

том, если на обоих коммутаторах используется функция STP UplinkFast, то восстановление каналов происходит почти мгновенно.

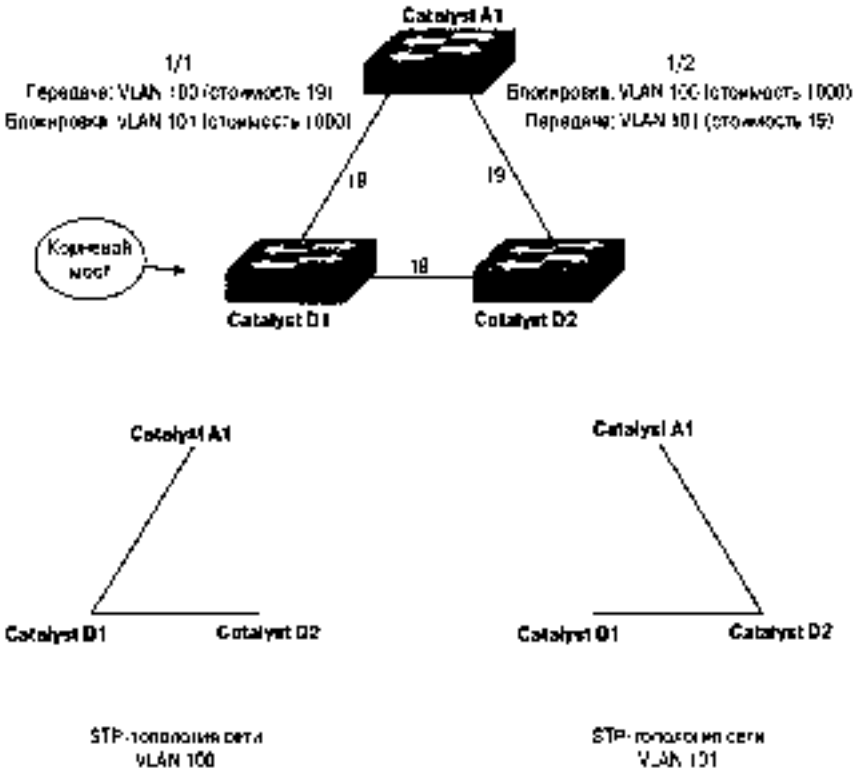


Рис. 7.3. Структура сети в примере балансировки нагрузки с помощью протокола STP

Коммутатор Catalyst D1 конфигурируется в качестве основного корневого моста для обеих VLAN-сетей, тогда как Catalyst D2 — в качестве дополнительного корневого моста. Если D1 выйдет из строя, корневым мостом станет коммутатор D2.

Catalyst D1 можно настроить с помощью следующих команд (если они доступны в операционной системе коммутатора).

Система COS	`set spantree root 100,101`

Система IOS	`spanning-tree vlan 100 root primary`
	`spanning-tree vlan 101 root primary`
	(обе команды вводятся в режиме глобальной конфигурации)

Альтернативное решение сводится к явной установке приоритетов мостов с помощью приведенных ниже команд (доступны на всех моделях коммутаторов Catalyst).

Система COS	`set spantree priority 8192 100` `set spantree priority 8192 101`

Система IOS	`spanning-tree vlan 100 priority 8192`
	`spanning-tree vlan 101 priority 8192`
	(обе команды вводятся в режиме глобальной конфигурации)

Catalyst D2 можно настроить для работы в качестве дополнительного корневого моста с помощью следующих команд.

Система COS	`set spantree root secondary 100,101`

Система IOS	`spanning-tree vlan 100 root secondary`
	`spanning-tree vlan 101 root secondary`
	(обе команды вводятся в режиме глобальной конфигурации)

В качестве альтернативы можно явно задать приоритеты моста для коммутатора D2.

Система COS	`set spantree priority 8200 100` `set spantree priority 8200 101`

Система IOS	`spanning-tree vlan 100 priority 8200`
	`spanning-tree vlan 101 priority 8200`
	(обе команды вводятся в режиме глобальной конфигурации)

Наконец, настраивается стоимость портов 1/1 и 1/2 для двух VLAN-сетей. Стандартная стоимость порта согласно диаграмме равна 19. Чтобы блокировать нежелательные маршруты, следует установить стоимость соответствующих портов равной 1000. Например, сеть VLAN 101 на порту 1/1 будет заблокирована, поскольку данный порт имеет наибольшую стоимость - 1000.

Система COS	`set spantree portvlancost 1/1 cost 1000 101` `set spantree portvlancost 1/2 cost 1000 100`

Система IOS	`interface fastethernet 1/1`
	`spanning-tree vlan 101 cost 1000`
	`interface fastethernet 5`
	`spanning-tree vlan 100 cost 1000`

7.3: точная настройка конвергенции распределенного связующего дерева

- Работа протокола STP основана на нескольких таймерах. Обычно стандартные значения таймеров используются для обеспечения корректного режима работы протокола. Стандартные настройки основаны на диаметре сети, состоящей из семи коммутаторов, но их можно регулировать для достижения большей скорости сходимости.
  - Hello-таймер (Hello Timer) инициирует периодическое hello-сообщения соседним коммутаторам.
  - Таймер задержки передачи (Forward Delay timer) определяет период времени, в течение которого порт остается в каждом из состояний прослушивания и самообучения.
  - Таймер старения (MaxAge timer) определяет время хранения BPDU-блока, полученного на назначенном порту. По истечении этого времени другие порты могут стать назначенными.
- Ожидается получение BPDU-блоков с постоянными интервалами. Если BPDU-блоки задерживаются на время, большее STP-таймера, то возможно ошибочное

появление изменений топологии. Такое состояние может быть зарегистрировано с помощью функции обнаружения *искажений BPDU-сообщений (BPDU skewing)*.

- Функция *STP PortFast* (механизм ускоренного включения) позволяет портам, подключающимся к узлам или конечным сетевым устройствам, входить в режим передачи сразу же после установки канала. Эта функция позволяет обходить обычные STP состояния порта для ускоренной загрузки, но также потенциально допускает формирование мостовых петель.
- Функция *STP UplinkFast* (механизм ускоренного переключения вышестоящих каналов) используется только на краевых коммутаторах (leaf-node switches — окончания ветвей распределенного связующего дерева), обычно расположенных на уровне доступа. Коммутатор отслеживает все потенциальные маршруты к корневому мосту, находящиеся в состоянии блокировки.
  - Когда корневой порт выходит из строя, альтернативный порт переводится в состояние передачи без последовательного прохождения обычных STP-состояний и задержек.
  - Когда включена функция UplinkFast, приоритет моста возрастает до 49152, что делает маловероятным переход узла в режим корневого моста. Стоимость каждого порта коммутатора увеличивается на 3000, поэтому они не будут выбраны в качестве корневых портов.
  - Когда активизируется альтернативный корневой порт, коммутатор объявляет вышестоящим коммутаторам о новом расположении нижестоящих устройств. Получателю с адресом 01-00-0C-CD-CD-CD, в мостовой таблице которого содержатся MAC-адреса станций, отправляются фиктивные многоадресные фреймы.
- Функция *STP BackboneFast* (механизм ускоренного переключения магистральных каналов) при неполном сбое вынуждает коммутаторы в основе сети активно искать альтернативные маршруты к корневому мосту.
  - Если эта функция используется, то она должна быть включена на всех коммутаторах сети. Коммутаторы для определения стабильности корневого маршрута используют механизмы запрос-ответ, поэтому все они должны иметь возможность участвовать.
  - Функция BackboneFast способна только сократить задержку конвергенции со стандартных 50 (20 секунд для таймера старения и по 15 секунд в состояниях прослушивания и самообучения) до 30 секунд.

Конфигурирование параметров настройки STP-конвергенции

1. Настройка стандартных STP-таймеров для регулировки сходимости *(необязательно)*.

Внимание!

Значения STP-таймеров нужно модифицировать только на корневом мосту, который затем посредством своих конфигурационных BPDU-блоков распространит новые значения среди остальных коммутаторов.

Если необходимо регулировать STP-таймеры, следует рассмотреть установку диаметра сети на корневом мосту распределенного связующего дерева. После установки диаметра значения всех остальных STP-таймеров вычисляются и регулируются автоматически.

Более подробная информация приведена в описании этапа 4 раздела "7.2: конфигурирование протокола STP".

а) Регулировка hello-таймера (*необязательно*).

Система COS	PVST+ `set spantree hello` *interval [vlan-list]* MISTP `set spantree hello` *interval* `mistp-instance` *instances* MST: `set spantree hello` *interval* `mst`
Система IOS	PVST+: `spanning-tree [vlan` *vlan*`] hello-time` *interval* (режим глобальной конфигурации)

Hello-таймер можно установить равным значению *interval* (от 1 до 10 секунд, стандартно — 2 секунды). Таймер можно настроить для определенных VLAN-сетей или STP-экземпляров либо глобально для сети VLAN 1 (COS) или для всех VLAN-сетей (IOS), если номер VLAN не указан.

б) Регулировка таймера задержки передачи (*необязательно*)

Система COS	PVST+: `set spantree fwddelay` *delay [vlans]* MISTP: `set spantree fwddelay` *delay* `mistp-instance` *[instances]* MST: `set spantree fwddelay` *delay* `mst`
Система IOS	PVST+: `spanning-tree vlan` *vlan* `forward-time` *delay* (режим глобальной конфигурации)

Интервал задержки передачи можно установить с помощью параметра *delay* (от 4 до 30 секунд, стандартно — 15 секунд) для определенных VLAN-сетей, определенных экземпляров либо глобально для VLAN 1 (COS) или всех VLAN-сетей (IOS), если номер VLAN-сети не задан.

в) Регулировка таймера старения (*необязательно*).

Система COS	PVST+: `set spantree maxage` *agingtime [vlan]* MISTP: `set spantree maxage` *agingtime* `mistp-instance` *instances* MST: `set spantree maxage` *agingtime* `mst`
Система IOS	PVST+: `spanning-tree [vlan` *vlan*`] max-age` *agingtime* (режим глобальной конфигурации)

Таймер старения может быть установлен равным значению *agingtime* (от 6 до 40 секунд, стандартно — 20 секунд) для определенных VLAN-сетей, экземпляров или глобально для VLAN 1 (COS) или всех VLAN-сетей (IOS), если номер VLAN-сети не задан.

2. Обнаружение проблем, связанных с искажением BPDU-сообщений (*необязательно*).

Система COS	`set spantree bpdu-skewing {enable | disable}`
Система IOS	Нет

Для поиска и устранения неисправностей можно отслеживать задержку или "искажение" получения BPDU-блоков. Эта функция стандартно отключена и включается с помощью ключевого слова **enable**. Коммутатор протоколирует любые возникающие BPDU-искажения. Собранную статистику впоследствии можно просмотреть с помощью команды **show spantree bpdu-skewing** *vlan [mod/port]*.

3. Использование функции PortFast STP-конвергенции для узлов уровня доступа (*необязательно*).

   а) Использование функции PortFast на определенных портах.

Система COS	**set spantree portfast** *mod/port* **{enable [trunk] | disable | default}**
Система IOS	**spanning-tree portfast** (режим конфигурирования интерфейса)

Можно включить (**enable**) или отключить (**disable**) функцию PortFast на немагистральных портах. Также можно использовать ключевое слово **trunk** для активации данной функции на магистральном канале. Ключевое слово **default** возвращает порт в стандартный режим работы, построенный с помощью необязательной команды **set spantree global-default portfast {enable | disable}**.

Внимание!

Включение функции PortFast на порту также предотвращает генерирование TCN BPDU-блоков в связи с изменением состояния порта. Несмотря на то что протокол STP продолжает функционировать на этом порту, препятствуя возникновению мостовых петель, изменения топологии не генерируются, когда подключенный узел переходит в активное или неактивное состояние.

Использовать функцию PortFast следует только на тех портах коммутатора, к которым подключены отдельные узлы. Иными словами, эту функцию не рекомендуется включать на портах коммутатора, которые подключены к другим коммутаторам или концентраторам, независимо от того, являются эти порты магистральными или нет.

   б) Включение функции PortFast BPDU Guard для повышения стабильности структуры STP (*необязательно*).

Система COS	**set spantree portfast bpdu-guard** *mod/port* **{enable | disable | default}**
Система IOS	**spanning-tree portfast bpduguard** (режим глобальной конфигурации)

Служба BPDU Guard при обнаружении BPDU-блока переводит немагистральный порт с включенной функцией PortFast в состояние `errdisable`. Для управления фильтрацией BPDU на порту используются ключевые слова **enable** и **disable**. Ключевое слово **default** используется для возврата порта в стандартный режим работы, глобально установленный необязательной командой **set spantree global-default bpdu-guard {enable | disable}**.

В операционной системе Supervisor IOS служба BPDU Guard включена глобально на всех портах с включенной функцией PortFast

Внимание!

Эта команда операционной системы Supervisor IOS недоступна на коммутаторах семейства Catalyst 2900XL и 3500XL.

в) Включение PortFast-фильтрации сообщений BPDU для остановки обработки блоков BPDU на порту *(необязательно)*.

Система COS	`set spantree portfast bpdu-filter mod/port {enable disable default}`
Система IOS	Нет

Функция BPDU-фильтрации приводит к тому, что коммутатор прекращает отправку BPDU-блоков на определенный порт. Входящие сообщения BPDU также не будут обрабатываться на этом порту. Для управления фильтрацией BPDU на порту используются ключевые слова `enable` и `disable`. Ключевое слово `default` служит для возврата порта в стандартный режим работы, глобально установленный необязательной командой `set spantree global-default bpdu-filter {enable | disable}`.

4. Использование функции UplinkFast STP-конвергенции для внешних каналов уровня доступа *(необязательно)*.

Система COS	`set spantree uplinkfast {enable | disable} [rate` *station_update_rate*`] [all-protocols {off | on}]`
Система IOS	`spanning-tree uplinkfast [max-update-rate` *packets-per-second*`]` (режим глобальной конфигурации)

Коммутатор способен генерировать фиктивные многоадресатные фреймы на скорости, определяемой параметром *station_update_rate*, который указывает количество фреймов, проходящих за 100 миллисекунд (в системе Catalyst OS стандартно 15 на 100 миллисекунд) или со скоростью, заданной значением *packets-per-second*, в пакетах в секунду (в Supervisor IOS стандартное значение равно 150 пакетов в секунду). Для генерации многоадресатных рассылок для каждой группы фильтрации протоколов используется ключевое слово `all-protocols`.

Совет

Для отключения функции UplinkFast и возврата приоритета места и стоимости портов в их исходные значения используется COS-команда `clear spantree uplinkfast`.

Если в коммутаторах Catalyst 2900XL, 3500XL и 3550 применяются стековые GBIC-модули (stacking modules) GigaStack, то к одному GBIC-порту *(Gigabit Interface Converter — конвертер гигабитового интерфейса)* можно подключить несколько коммутаторов. Стековый GBIC-конвертер, обладающий двумя физическими соединениями, становится многоканальной магистралью, что делает функцию UplinkFast неэффективной при изменениях топологии.

В дополнение к механизму UplinkFast можно включить функцию CSUF *(Cross-Stack UplinkFast — межстековая функция UplinkFast)*, которая позволяет быстро обходить обрывные участки через стековое GBIC-соединение. Эту функцию можно включить только на одном стековом GBIC интерфейсе коммутатора. Вместе с тем ее следует включить на всех коммутаторах, подключенных к стеку.

Система COS	Нет
Система IOS	`spanning-tree stack-port` (режим конфигурирования интерфейса)

5. Использование функции BackboneFast STP-конвергенции для избыточных магистральных каналов *(необязательно)*.

Система COS	`set spantree backbonefast {enable | disable}`
Система IOS	`spanning-tree backbonefast` (режим глобальной конфигурации)

Если вы хотите применить эту функцию, ее следует включить на всех коммутаторах в сети. Служба BackboneFast включается или отключается для всех VLAN-сетей на коммутаторе.

7.4: навигация по топологии распределенного связующего дерева

Несмотря на то что навигация по топологии распределенного связующего дерева довольно утомительна, она, как правило, представляет собой единственный способ проверить, работает ли протокол STP именно так, как планировалось. Часто встречаются диаграммы коммутируемой сети, описывающие физические или логические соединения. В то же время топология распределенного связующего дерева обычно остается недокументированной до тех пор, пока не возникнет какая-либо проблема.

Может появиться необходимость определить и устранить неисправность в незнакомой сети или сети, полная документация которой отсутствует. В таком случае нужно создать план активной в текущий момент STP-топологии, особенно следует определить расположение корневого моста.

1. Указание местоположения корневого моста.

   а) Выбор коммутатора, используемого в качестве начальной точки.

   В идеале желательно начать с корневого моста на "вершине" STP-иерархии. Если неизвестно, какой коммутатор является корневым для данной VLAN-сети, то в качестве начальной точки можно выбрать любой.

   б) Отображение корневого идентификатора (Root ID), локального BID-идентификатора и корневого порта.

Система COS	`show spantree vlan active`
Система IOS	`show spanning-tree vlan vlan` (режим привилегированного или обычного пользователя)

Ниже приводится пример информации, отображаемой операционной системой COS. Следует заметить, что назначенный корневой порт состоит из списка портов коммутатора (1/1, 1/2, 2/1 и 2/2). В данном COS-коммутаторе четыре порта объединены в EtherChannel-канал, который рассматривается протоколом STP как один логический канал.

```
switch (enable) show spantree 534 active
VLAN 534
Spanning tree mode          PVST+
Spanning tree type          ieee
Spanning tree enabled
Designated Root             00-d0-0a-57-3e-15
Designated Root Priority    8000
Designated Root Cost        2
Designated Root Port        1/1-2,2/1-2 (agPort 11/1)
Root Max Age 20 sec         Hello Time 2 sec Forward Delay 15 sec
Bridge ID MAC ADDR              00-d0-ff-6a-2a-15
Bridge ID Priority              32768
Bridge Max Age 20 sec       Hello Time 2 sec Forward Delay 15 sec
Port                        Vlan  Port-State     Cost      Prio
Portfast Channel_id
------------------------   ----  ------------ --------- ---- -----
1/1-2,2/1-2                 534   forwarding            2   32
disabled 769
3/1,4/1                     534   forwarding           12   32
disabled 833
5/4                         534   forwarding            4   32
disabled 0
5/5                         534   forwarding            4   32
disabled 0
5/6                         534   forwarding            4   32
disabled 0
5/7                         534   forwarding            4   32
disabled 0
```

Ниже приводится пример выполнения указанной команды для системы Supervisor IOS.

```
switch#show spanning-tree vlan 534

Spanning tree 534 is executing the IEEE compatible Spanning Tree
protocol
  Bridge Identifier has priority 49152, address 0005.5225.45ef
  Configured hello time 2, max age 20, forward delay 15
  Current root has priority 8000, address 00d0 3a57.3a15
  Root port is 67, cost of root path is 3000
  Topology change flag not set, detected flag not set, changes 172
  Times:  hold 1, topology change 35, notification 2
          hello 2, max age 20, forward delay 15
  Timers: hello 0, topology change 0, notification 0
  Fast uplink switchover is enabled
  Shark port is GigabitEthernet5/2
Interface Fa6/1 (port 1) in Spanning tree 534 is FORWARDING
   Port path cost 3010, Port priority 128
   Designated root has priority 8000, address 00d0.0457.3a10
```

```
   Designated bridge has priority 49152, address 0005.3225.45ef
   Designated port is 14, path cost 3006
   Timers: message age 0, forward delay 0, hold 0
   BPDU: sent 2967446, received 0
   The port is in the portfast mode
...[output removed]...
Interface Gi0/1 (port 67) in Spanning tree 534 is FORWARDING
   Port path cost 3004, Port priority 128
   Designated root has priority 8000, address 00d0.0457.3a15
   Designated bridge has priority 32768, address 00d0.ff8a.2a15
   Designated port is 7, path cost 2
   Timers: message age 3, forward delay 0, hold 0
   BPDU: sent 1, received 2967537
Interface Gi0/2 (port 75) in Spanning tree 534 is FORWARDING
   Port path cost 4, Port priority 128
   Designated root has priority 8000, address 00d0.0457.3a15
   Designated bridge has priority 49152, address 0005.3225.45ef
   Designated port is 75, path cost 2006
   Timers: message age 0, forward delay 0, hold 0
   BPDU: sent 2967519, received 1
switch#
```

в) Переход от корневого порта к корневому мосту

Напомним, что в коммутаторе имеется только один корневой порт и этот порт ведет к корневому мосту. Коммутатор может иметь множество назначенных портов, которые ведут от корневого моста. Нужно найти соседний коммутатор, который подключен к корневому порту.

Следует отметить, что в примере отображаемой информации (этап 1,б) система COS показывает корневой порт как физический модуль и номер порта (1/1, 1-2, 2/1-2 — как один объединенный EtherChannel-канал), а операционная система Supervisor IOS отображает его как номер логического порта (порт 67). Номер порта является индексом интерфейсов согласно STP. Можно либо постранично просматривать отображаемую информацию до тех пор, пока не встретится интерфейс с номером этого порта, либо, используя EXEC-команду **show spanning-tree brief | begin VLAN** *vlan*, отобразить только номер порта, связанного с определенной VLAN-сетью. Рассмотрим пример:

```
switch# show spanning-tree brief | begin VLAN534
VLAN534
  Spanning tree enabled protocol IEEE
  ROOT ID     Priority 8000
              Address 00d0.0457.3a15
              Hello Time   2 sec  Max Age 20 sec ForWard Delay
              15 sec
  Bridge ID   Priority    49152
              Address     0005.3225.45ef
              Hello Time   2 sec  Max Age 20 sec Forward Delay
              15 sec
Port                              Designated
Name      Port ID Prio Cost Sts  Cost  Bridge ID       Port ID
--------- ------- ---- ---- ---  ----  --------------  -------
Fa3/1     128.13  128  3100 BLK  3006  0005.3225.45ef  128.13
Fa3/2     128.14  128  3019 FWD  3006  0005.3225.45ef  128.14
```

```
...[output removed]...
Gi0/1    128.67   128   3004 FWD  2      00d0.ff9a.2a15  128.7
Gi0/2    128.75   128   4    FWD  3006   0005.3255.45ef  128.75
```

В данном случае STP-порт 67 соответствует физическому интерфейсу GigabitEth. Операционная система SuperStack IOS также отображает дополнительную информацию — MAC-адрес назначенного моста на корневом порту.

г) Идентификация назначенного моста на корневом порту.

Система COS	`show cdp neighbor` *mod/num* **detail**
Система IOS	**show cdp neighbor** *type mod/num* **detail** (режим привилегированного или обычного пользователя)

Соседний коммутатор может быть найден с помощью протокола обнаружения устройств Cisco *(Cisco Discovery Protocol — CDP)*, если он используется. IP-адрес соседнего коммутатора следует искать в отображаемой информации, пример которой приводится ниже.

```
switch#show cdp neighbor gigabitEthernet 0/1 detail
-------------------------
Device ID: SCA03320048(Switch-3)
Entry address(es):
  IP address: 192.168.254.17
Platform: WS-C6509, Capabilities: Trans-Bridge Switch
Interface: GigabitEthernet0/1, Port ID (outgoing port): 5/7
Holdtime : 120 sec
```

После вычисления IP-адреса можно открыть Telnet-сеанс с соседним коммутатором.

д) Следует повторять этапы от 1,б до 1,г до тех пор, пока корневой мост не будет найден.

Каким образом пользователь может узнать, что он достиг корневого моста? Локальный BID-идентификатор будет равен идентификатору корневого моста, а стоимость порта будет равна нулю. Ниже приведен пример информации, отображаемой коммутатором под управлением операционной системы Catalyst OS.

```
switch (enable) show spantree 534 active
VLAN 534
Spanning tree mode          PVST+
Spanning tree type          ieee
Spanning tree enabled
Designated Root             00-d0-04-57-3a-15
Designated Root Priority    9000
Designated Root Cost        0
Designated Root Port        1/0
Root Max Age  20 sec   Hello Time 2 sec    Forward Delay 15 sec
Bridge ID MAC ADDR          00-d0-04-57-3a-15
Bridge ID Priority          8000
Bridge Max Age 20 sec    Hello Time 2 sec    Forwrd Delay 15 sec
Port                     Vlan Port-State    Cost      Prio PortFast
Channel_id
------------------------ ---- ------------- ----- ---- -------- ---
1/1-2,2/1-2              534  forwarding        2   32 disabled 769
```

```
3/1-2,4/1-2                     554  forwarding          2    32 disabled 997
3/3                             554  forwarding          4    32 disabled 2
```

Отметим, что локальный MAC-адрес моста и MAC-адрес корневого моста, как и значения приоритета, совпадают. (Помните, что BID-идентификатор состоит из MAC адреса и значения приоритета моста.) Рассматриваемый коммутатор является корневым мостом для сети VLAN 554.

2. Создание схемы активной топологии "сверху вниз".

   Начиная с корневого моста, определим месторасположение остальных коммутаторов, входящих в состав распределенного связующего дерева для определенной VLAN-сети.

   а) Идентификация соседних коммутаторов.

Система COS	`show cdp neighbor detail`
Система IOS	`show cdp neighbor detail` (режим привилегированного или обычного пользователя)

   Все соседи могут быть идентифицированы по имени, IP-адресу и соединяющим портам. Обычно на коммутаторах, расположенных вблизи от основного уровня, отображается больший список соседних коммутаторов, а на коммутаторах уровня доступа — меньший.

   б) Определение BID-идентификатора, корневого и назначенных портов, а также их стоимости

Система COS	`show spantree vlan active`
Система IOS	`show spanning-tree brief | begin VLAN vlan` (режим привилегированного или обычного пользователя)

   BID-идентификатор и корневой порт отображаются в первую очередь. Порты коммутатора во VLAN-сети с номером `vlan` перечисляются наряду с их STP-состояниями и стоимостью. Назначенными портами являются те, которые отмечены как находящиеся в состоянии *forwarding*.

   в) Идентификация заблокированных портов.

Система COS	`show spantree blockedports vlan`
Система IOS	`show spanning-tree vlan vlan | include BLOCKING` (режим привилегированного или обычного пользователя)

   г) Переход к соседнему коммутатору и повторение этапов 2.а–2.в.

Дополнительная литература

Рекомендуемые ниже источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Келледи Кларк, Кевин Гамильтон. *Принципы коммутации в локальных сетях Cisco*, Вильямс, 2003.

Karen Webb. *Building Cisco Multilayer Switched Networks*, Cisco Press

Tim Boyles and David Hucaby, *CCNP Switching Exam Certification Guide*, Cisco Press.

Radia Perlman, *Interconnections: Bridges, Routers, Switches and Internetworking Protocols*, Addison-Wesley.

Основные понятия и конфигурирование функции Cisco Uplink Fast (Understanding and Configuring the Cisco Uplink Fast Feature): www.cisco.com/warp/customer/473/51.html.

Основные понятия функции Backbone Fast протокола распределенного связующего дерева (Understanding Spanning Tree Protocol's Backbone Fast Feature): www.cisco.com/warp/customer/473/19.html.

MAC-мосты IEEE-стандарта 802.1D (802.1D MAC Bridges, IEEE): www.ieee802.org/1/pages/802.1D.html.

Множественные распределенные связующие деревья IEEE-стандарта 802.1s, (802.1s Multiple Spanning Trees, IEEE): www.ieee802.org/1/pages/802.1s.html.

В этой главе...

- 8.1: **многоуровневая коммутация.** В этом разделе описана настройка коммутации третьего уровня на платформах, поддерживающих технологию *многоуровневой коммутации (MultiLayer Switching — MLS)* и имеющих отдельные блоки процессора маршрутизации и коммутации.
- 8.2: **экспресс-коммутация Cisco.** В данном разделе описаны этапы настройки коммутации третьего уровня на платформах, поддерживающих функцию *экспресс-коммутации корпорации Cisco (Cisco Express Forwarding — CEF)*.
- 8.3: **экспорт данных NetFlow.** В разделе описана методика использования функции *экспорта данных NetFlow (NetFlow Data Export — NDE)* для отправки статистики потоков данных третьего уровня с целью сбора и анализа.
- 8.4: **резервирование модулей MSFC в одном устройстве.** В этом разделе описана методика конфигурирования двойных модулей *функциональных плат многоуровневой коммутации (Multilayer Switching Feature Card — MSFC)* для выполнения операций восстановления после сбоев, когда активна только одна плата.
- 8.5: **MSFC-резервирование с синхронизацией конфигурации.** В разделе поэтапно описана активизация сдвоенных модулей MSFC при поддержке общей конфигурации.
- 8.6: **резервирование маршрутизаторов с помощью протокола HSRP.** В этом разделе приведено описание этапов конфигурирования коммутаторов третьего уровня для совместного использования ими IP-адреса в целях обеспечения резервирования шлюзов.

Глава 8

Многоуровневая коммутация

В табл. 8.1 перечислены коммутирующие платформы Cisco Catalyst, поддерживающие коммутацию третьего уровня. Чтобы настроить коммутацию третьего уровня, необходимо найти в таблице конкретную модель коммутатора и тип поддерживаемой им коммутации третьего уровня (MLS или CEF), а затем перейти к соответствующему разделу.

Таблица 8.1. Коммутирующие платформы Catalyst с возможностями коммутации третьего уровня

Модель коммутатора Catalyst	Модуль	Поддерживаемый тип коммутации третьего уровня	Раздел
3500XL	Нет	Нет	Нет
3550	Нет	Аппаратная функция CEF	8-2
4000	Sup II	MLS[1]	Нет
4000	Sup III	Аппаратная функция CEF	8-2
4908G-L3	Нет	Аппаратная функция CEF	8-2
5000	Sup IIg	MLS[1]	Нет
	Sup III + NFFC	MLS[1]	Нет
6000	Без платы MSFC	Нет	Нет
	Плата MSFC + функциональная плата политики (Policy Feature Card — PFC)	Внутренняя MLS-коммутация, используемая автоматически	8-1
	MSFC2+PFC2	Аппаратная CEF-коммутация, используемая автоматически	8-2

[1] *Технология MLS поддерживается с помощью внешнего маршрутизатора или установленного модуля маршрутизации третьего уровня (RSM, RSFC или модуля служб третьего уровня). Коммутатор (SE) и маршрутизатор (RP) необходимо конфигурировать независимо.* — Прим. ред.

8.1: многоуровневая коммутация

- В *многоуровневой коммутации (Multilayer switching — MLS)* обработка третьего уровня осуществляется путем комбинирования отдельных функций маршрутизации и коммутации на различных модулях коммутатора.
- Технология MLS поддерживается на коммутаторе Catalyst 6000 с помощью плат MSFC (процессор маршрутизации) и PFC (блок коммутации третьего уровня). На платформах Catalyst 4000 и 5000 в качестве процессора маршрутизации может использоваться внешний маршрутизатор.
- Технология MLS способна осуществлять коммутацию третьего уровня для IP-трафика, многоадресатного IP-трафика (*IP Multicast MLS*, или *MMLS*) и IPX-трафика.
- Процесс многоуровневой коммутации включает в себя несколько этапов.
  - *Процессор маршрутизации* (RP) определяет маршрут первого пакета в потоке данных.
  - *Блок коммутации* (SE) на основании первого пакета (пакета "кандидата") создает в MLS-кэше запись для потока.
  - Когда блок коммутации получает возвращаемый от RP-процессора пакет, запись MLS-кэша дополняется информацией об отправителе и получателе. В период прохождения этого потока последующие пакеты коммутируются в SE-блоке.
  - При передаче пакетов блок коммутации также перезаписывает MAC-адреса отправителя и получателя, время существования *(time-to-live — TTL)* IP-пакетов и значения контрольных сумм второго и третьего уровней. Эти операции выполняются аппаратным обеспечением и подобны процессам, которые выполняет традиционный маршрутизатор при перенаправлении пакетов.
  - Запись MLS-кэша для потока удаляется, когда соединение закрывается, или по истечении времени старения.
- Технология MLS создает свой кэш потоков данных на основании следующих сведений:
  - **для IP-трафика** — адреса получателя, адреса отправителя и получателя или адреса получателя и отправителя и номера их портов ("портный поток").
  - **для многоадресатного IP-трафика** — адрес отправителя, номер VLAN-сети отправителя и многоадресатной группы получателя.
  - **для IPX-трафика** — адрес получателя.
- Технология MLS с помощью функции NDE способна создавать статистические отчеты о потоках данных. Более подробная информация приведена в разделе "8.3: экспорт данных NetFlow".

Конфигурирование функции

1. Настройка функции MLS на RP-процессоре *(необязательно)*.
   a) Конфигурирование функции MLS *(необязательно; только для внешних процессоров маршрутизации)*.

Совет

В этой главе MLS рассматривается как функция, интегрированная между блоком коммутации и процессором маршрутизации в коммутаторе Catalyst 6000 с платами MSFC и PFC. Функция MLS автоматически включается между модулями RP и SE, поэтому этап 1,а не является обязательным.

Начинать с этапа 1,а следует при конфигурировании RP-части MLS на коммутаторах Catalyst 4000 (блок Supervisor I или II) и Catalyst 5000, которые требуют наличия внешнего процессора маршрутизации.

- Включение функции MLS на маршрутизаторе:

Система COS	Нет
Система IOS	`mls rp ip` (режим глобальной конфигурации)

- Идентификация VTP-домена (*необязательно*).

Система COS	Нет
Система IOS	`mls rp vtp-domain name` (режим конфигурирования интерфейса)

Если процессор маршрутизации обменивается пакетами с коммутатором в VTP-домене, необходимо указать имя (*name*) этого домена. Впоследствии RP-процессор может получать сведения о конфигурации VLAN-сети от VTP-сервера.

- Указание номера VLAN для сети или же ISL-интерфейсов (*необязательно*)

Система COS	Нет
Система IOS	`mls rp vlan-id vlan-id` (режим конфигурирования интерфейса)

- Включение интерфейса для MLS-управления.

Система COS	Нет
Система IOS	`mls rp management-interface` (режим конфигурирования интерфейса)

Этот интерфейс используется для отправки и получения управляющей MLS-информации (MLSP-пакетов). Следует использовать VLAN-интерфейс, который подключен к коммутаторам, использующим MLS. Часто в качестве такого экземпляра выступает административная VLAN-сеть.

б) Использование однонаправленной MLS-коммутации на определенных VLAN-интерфейсах (*необязательно*)

Система COS	Нет
Система IOS	`[no] mls {ip ipx}` (режим конфигурирования интерфейса)

Стандартно одноадресатная IP MLS-коммутация на всех интерфейсах включена, а IPX-служба MLS отключена. Для отключения многоуровневой коммутации на интерфейсе используется ключевое слово `no`.

в) Указание маски потока (*необязательно*)

Система COS	Нет

Система IOS	`mls flow ip {destination | destination-source | full}`
	(режим глобальной конфигурации)
	или
	`mls flow ipx {destination | destination-source}`
	(режим глобальной конфигурации)

Маска потока MLS может быть установлена с помощью адресов получателей (`destination` — стандартная установка), адресов отправителя и получателя (`destination-source`) или адресов и номеров портов отправителя и получателя (`full`).

г) Исключение из многоуровневой коммутации определенных протоколов (*необязательно*).

Система COS	Нет
Система IOS	`mls exclude protocol {tcp | udp | both} [port port-number]` (режим глобальной конфигурации)

Стандартно все протоколы и номера портов используются для создания записей потоков в MLS-кэше. Можно исключить определенные протоколы. Для этого указывается название протокола (`tcp`, `udp` или `both` — протоколы TCP и UDP) и номер порта (`port-number` от 1 до 65535).

2. **Настройка технологии MLS на SE-блоке** (*необязательно*).

а) Включение MLS-коммутации для протоколов IP и IPX (*необязательно, только для коммутаторов Catalyst 5000*)

Системы COS	`set mls {enable | disable} [ip | ipx]`
Система IOS	Нет

Стандартно MLS-коммутация для протокола IP включена, а для протокола IPX — отключена.

б) Идентификация внешних процессоров маршрутизации MLS (*только для устройств Catalyst 5000*).

Системы COS	`set mls include {ip | ipx} ip-addr1 [ip-addr2...]`
Система IOS	Нет

Внешние маршрутизаторы, которые работают с коммутатором для MLS, имеют IP-адреса *ip-addr1*, *ip-addr2* и т.д. Маршрутизаторы должны подключаться к коммутатору через один магистральный порт. Модули маршрутизаторов, такие, как RSM и RSFC, интегрированные в шасси коммутаторов Catalyst 5000, добавляются как процессоры маршрутизации автоматически.

в) Настройка периода старения MLS-кэша (*необязательно*).

Система COS	`set mls agingtime {ip | ipx} agingtime` `set mls agingtime fast` *fastagingtime pkt-threshold* `set mls agingtime long-duration` *longagingtime*
Система IOS	Нет

Созданная MLS-запись содержится в кэше в течение периода старения (*agingtime* от 8 до 2024 секунд, кратно 8, стандартное значение равно 256 секундам), если с ее помощью эти записи пакеты не обрабатываются.

С помощью ключевого слова **fast** можно ускорить удаление из кэша MLS-записей для очень коротких потоков, таких, как, например, DNS-запросы. Запись устаревает, если с ее помощью в течение периода быстрого старения (*fastagingtime*) было обработано не больше пакетов, чем указано в параметре *pkt-threshold*. Период быстрого старения (*fastagingtime*) устанавливается равным от 0 до 128 секунд, его значение кратно 8, стандартно оно равно нулю или не используется. Пороговое значение (*pkt-threshold*) — это количество пакетов (0, 1, 3, 7, 15, 31, 63 или 127), стандартно оно равно нулю.

Записи MLS-кэша для активно коммутируемых потоков также могут устареть, до того как станут неактивными. Такой режим задается с помощью ключевого слова **long-duration**. После создания MLS-записи она удаляется по истечении периода, равного *longagingtime* (от 64 до 1920 секунд, кратно 64, стандартное значение равно 1920 секундам).

Совет

Согласно рекомендациям корпорации Cisco, количество записей в MLS-кэше не должно превышать 32 тысяч. Потоки, превышающие размер кэша, как правило, отправляются RP-процессору для обычной маршрутизации, а не коммутируются посредством технологии MLS. Размер кэша можно отслеживать с помощью команд **show mls** (COS) и **show mls ip count** (IOS).

Если количество записей в кэше превышает 32 768, рекомендуется перейти к регулировке MLS-таймеров, начиная с уменьшения времени старения на 8 секунд до получения нормальных потоков. Размер кэша следует периодически контролировать. Если он все еще превышает 32 000, рекомендуется последовательно с шагом 64 секунды сокращать время старения.

В ситуации, когда могут возникнуть много коротких потоков, можно также настроить таймер быстрого старения. По умолчанию интервал таймера равен нулю или не используется вовсе. Начинать следует со значения 128 секунд. Если размер кэша превышает 32 000, необходимо уменьшить время быстрого старения.

г) Установка маски минимального MLS-потока (*необязательно*).

Система COS	`set mls flow {destination | destination-source full}`
Система IOS	Нет

В качестве маски потока, используемой для функции MLS, между устройствами RP и SE фактически выбирается наиболее специфическая или самая

длинная необходимая маска. Стандартно такие устройства используются в качестве минимальной маски получателя (destination). Если процессор маршрутизации был сконфигурирован для более длинной маски или обладает расширенными списками доступа, применяемыми к его интерфейсам, согласовывается более длинная маска.

3. Настройка *многоадресатной IP MLS-коммутации* на RP процессоре (*необязательно*).

   а) Включение многоадресатной IP-маршрутизации.

   - Запуск многоадресатной маршрутизации на RP-процессоре.

Система COS	Нет
Система IOS	`ip multicast-routing` (режим глобальной конфигурации)

   - Включение *независящего от протокола многоадресатного IP-вещания (Protocol Independent Multicast — PIM)* на каждом многоадресатном интерфейсе

Система COS	Нет
Система IOS	`ip pim {dense-mode | sparse-mode | sparse-dense-mode}` (режим конфигурирования интерфейса)

   Стандартно при многоадресатной маршрутизации используется разреженный режим (`sparse-dense-mode`) протокола PIM.

   б) Включение функции IP MMLS на процессоре маршрутизации

Система COS	Нет
Система IOS	`mls ip multicast` (режим глобальной конфигурации)

   Стандартно функция IP MMLS отключена, даже если активна функция многоадресатного IP-вещания (IP multicast). После включения функция MMLS используется только на интерфейсах с включенной многоадресатной IP PIM-маршрутизацией.

   в) Использование порогового значения для контроля уровня записи MMLS-кэша (*необязательно*).

Система COS	Нет
Система IOS	`mls ip multicast threshold` *pps* (режим глобальной конфигурации)

   Пороговое значение можно применять для предотвращения заполнения MMLS-кэша краткосрочными записями. Если скорость новых многоадресатных пакетов превышает порог *pps* (количество пакетов в секунду, от 10 до 10000, стандартного значения не существует), то пакеты коммутируется посредством службы MMLS. Если скорость ниже порогового значения, многоадресатные пакеты отправляются процессору маршрутизации для обычной (не MMLS) обработки.

4. Настройка функции IP MMLS в блоке коммутации (*необязательно*).

Совет

Если многоадресатная IP-маршрутизация включена, служба IP MMLS на стороне SE-блока включается автоматически. Полная информация по конфигурированию приведена в главе 9, "Многоадресатные службы".

Пример конфигурирования многоуровневой коммутации

MLS-коммутация конфигурируется на коммутаторе со встроенными модулями RP и SE (например, в коммутаторе Catalyst 6000 блок Supervisor I и платы PFC и MSFC). Интерфейсы сетей VLAN 100 и VLAN 200 конфигурируются на использование MLS для коммутации потоков данных третьего уровня. Согласно конфигурации RP-модуля необходимо использовать маску полного MLS-потока.

SE-часть конфигурируется на использование стандартного времени старения MLS-кэша (256 секунд). Однако время ускоренного старения регулируется таким образом, что записи кэша удаляются, если в течение восьми секунд периода ускоренного старения коммутируется более 31 пакета.

Многоадресатная IP-маршрутизация поддерживается с помощью функции IP MMLS на RP-процессоре. SE-часть автоматически настраивается на поддержку многоадресатной MLS-коммутации.

Система COS	`set mls agingtime fast 8 31`

Система IOS	`interface vlan 100`
	(режим глобальной конфигурации)
	`mls ip`
	(режим конфигурирования интерфейса)
	`interface vlan 200`
	(режим глобальной конфигурации)
	`mls ip`
	(режим конфигурирования интерфейса)
	`mls flow ip full`
	`ip multicast-routing`
	`mls ip multicast`
	(три последние команды вводятся в режиме глобальной конфигурации)

Отображение сведений по MLS-коммутации

В табл. 8.2 перечислены команды процессора маршрутизации (операционная система IOS) и команды блока коммутации (операционная система коммутатора COS), используемые для получения информации по MLS-коммутации.

Таблица 8.2. Команды для отображения MLS-информации

Функция отображения	Операционная система коммутатора	Команда
Состояние службы IP MLS	COS	`show mls [ip | ipx] [module]`

Продолжение табл. 8.2

Функция отображения	Операционная система коммутатора	Команда
	IOS	`show mls rp [ip | ipx  interface interface interface-number | vtp-domain domain]`
Информация по службе IP MLS	COS	`show mls`
	IOS	`show mls ip [any | destination {hostname ip-address} | detail | flow {tcp | udp} | {interface {interface interface-number}} {vlan vlan} | {macd destination-mac-address} | {macs source-mac-address} | {module number} | source {hostname | ip-address}]`

MLS-статистика	COS	`show mls statistics protocol`
		`show mls statistics entry [mod]`
		`show mls statistics entry ip [mod] [destination ip_addr_spec] [source ip_addr_spec] [protocol protocol [src-port src_port] [dst-port dst_port]]`

	IOS	`show mls statistics`
Протоколы, исключенные из службы IP MLS	COS	`show mls exclude protocol`
	IOS	Нет
Записи IP MLS-кэша	COS	`show mls entry ip [mod] [destination ip_addr_spec] [source ip_addr_spec] [protocol protocol] [src-port src_port] [dst-port dst_port] [short | long]`
	IOS	Нет
MLS-информация протокола IPX	COS	`show mls statistics entry ipx [mod] [destination ipx_addr_spec] [source ipx_addr_spec]`
	IOS	`show mls ipx [{destination ipx-network} | {interface {interface interface-number}} {vlan vlan-id} | {macd destination-mac-address} | {macs source-mac-address} | {module number} | {source {hostname | ipx-network}}] [detail]`
Записи протокола IPX MLS-кэша	COS	`show mls entry ipx [mod] [destination ipx_addr_spec] [short | long]`
	IOS	Нет
Размер MLS-кэша	COS	`show mls`

Функция отображения	Операционная система коммутатора	Команда
	IOS	`show mls ip count` `show mls ipx count`
Сведения о функции IP MMLS	COS	`show mls multicast` `show mls multicast statistics {mod}`
	IOS	`show mls ip multicast [{{connected | group} {hostname | ip-address} {ip-mask}} {interface {interface interface-number}} | {module number} | {source {hostname | ip-address}} | statistics | summary}]`
Записи протокола IP MMLS-кэша	COS	`show mls multicast entry {[all] {short | long]}` `show mls multicast entry {[mod] [vlan vlan_id] [group ip_addr]} [source ip_addr] [long | short]`
	IOS	Нет

8.2: экспресс-коммутация Cisco

- *Экспресс-коммутация Cisco (Cisco Express Forwarding — CEF)* обеспечивает аппаратную поддержку перенаправления всех пакетов в потоке данных.
- Технология CEF реализована в коммутаторах Catalyst серии 2948G-L3, 4908G-L3, 4000 Supervisor III и 3550. Кроме того, она доступна на коммутаторе Catalyst 6000 как совместная функция блока коммутации третьего уровня PFC2 и модуля процессора маршрутизации MSFC2.
- Процессор маршрутизации выполняет протоколы маршрутизации и заполняет таблицы нескольких видов.
  - **Обычная таблица маршрутизации.** Таблица маршрутов и получателей следующих транзитных переходов, определяемых протоколами маршрутизации, административными расстояниями, метриками и другими параметрами.
  - **База пересылочной информации (Forwarding Information Base — FIB).** В FIB-базе каждый известный маршрут предоставляется в виде древовидной иерархической структуры. Аппаратные средства могут быстро обнаружить маршруты с наибольшей длиной совпадения, указывающие на записи следующего перехода в таблице смежности.
  - **Таблица смежности (adjacency table).** В таблицу смежности вносятся все обнаруженные адреса маршрутизаторов следующих транзитных переходов и ARP-ответы *(Address Resolution Protocol — протокол преобразования адресов)*. Эта таблица позволяет осуществлять эффективный поиск информации передачи от третьего ко второму уровню.

- CEF-технология поддерживает высокопроизводительную коммутацию IP-, многоадресатного IP- и IPX-трафика
- С помощью CEF-коммутации можно коммутировать пакеты к одному получателю по нескольким (до шести) маршрутам с равной стоимостью.
- Для обеспечения прибытия пакетов на интерфейсы, являющиеся получателями маршрутами к отправителю, в CEF может применяться *методика проверки обратного маршрута (Reverse Path Forwarding — RPF)*. Она используется для обнаружения фиктивных или подложных адресов в получаемых пакетах при проявлении какой-либо злонамеренной активности.
- Многоадресатный IP-трафик коммутируется с помощью CEF только для многоадресатных групп внутри блоков 225.0.0.* через блок 239.0.0.* и внутри блока 224.128.0.* через блок 239.128.0.*. С помощью CEF-коммутации не отправляются данные в группу 224.0.0.*, поскольку эти адреса зарезервированы для протоколов маршрутизации и должны данному отправляться всем портам, которые являются переходными в какой-либо VLAN-сети.
- Когда процессор маршрутизации создает базу данных FIB, информация из нее загружается и используется аппаратным обеспечением блока коммутации. В коммутаторе Catalyst 6000 база данных FIB загружается из платы MSFC2 в модуль PFC2, а также в какую-либо из присутствующих *плат распределенной коммутации (Distributed Forwarding Cards — DFC)*
- В дополнение к CEF-таблицам независимо генерируется таблица коммутации NetFlow (идентичная таблице MLS). Она создается только для обеспечения учетной информации потоков данных и информация из нее может экспортироваться внешним приложениям. Более подробно об этом рассказывается в разделе "8.3: экспорт данных NetFlow"

Конфигурирование CEF-коммутации

Совет

CEF-коммутация автоматически включается на платформах, поддерживающих ее, и не может быть отключена

1. Использование метода RPF для обнаружения фиктивных или неверно сформированных пакетов *(необязательно)*.

Система COS	Нет
Система IOS	`ip verify unicast reverse-path [list]` (режим конфигурирования интерфейса)

При стандартных настройках функция RPF глобально отключена на коммутаторе. Ее можно включать на определенных VLAN-интерфейсах.

Для каждого пакета, принятого на данном интерфейсе, механизм CEF проверяет, присутствует ли в базе данных FIB достоверный маршрут обратно к адресу отправителя. В обратном маршруте в качестве возможного маршрута к отправителю необходимо использовать принимающий интерфейс. Если есть несколько имеющих равную стоимость маршрутов к отправителю, то все они являются достоверными.

Стандартно служба CEF отбрасывает все входящие пакеты, не прошедшие RPF-тест на интерфейсе. Чтобы определенным образом обусловить отбрасывание таких пакетов, можно задать стандартный или расширенный список IP-доступа (его имя или номер указывается в параметре *list*). Пакеты, соответствующие разрешающему (permit) условию, передаются, даже если не прошли RPF-тест, тогда как пакеты, удовлетворяющие запрещающему условию (deny), уничтожаются.

2. Настройка балансировки нагрузки CEF-коммутации *(необязательно; только для коммутаторов Catalyst 6000)*

Система COS	`set mls cef load-balance {full | source-destination-ip}`
Система IOS	Нет

Потоки данных распределяются по параллельным маршрутам согласно результатам хэш-функции, основанной на адресах отправителя и получателя (стандартная установка — IP-адрес отправителя `source-destination-ip`) или на их адресах и номерах портов (`full`). CEF не поддерживает балансировку нагрузки по отдельным пакетам.

3. Управление скоростью CEF-пакетов, возвращаемых процессору маршрутизации *(необязательно; только для коммутаторов Catalyst 6000)*.

Система COS	Нет
Система IOS	`mls ip cef rate-limit` *pps* (режим глобальной конфигурации)

Некоторые пакеты не могут быть полностью переданы посредством CEF-коммутации. Такие пакеты должны возвращаться процессору маршрутизации (MSFC2) для обработки. Они включают в себя пакеты, требующие ARP-запроса, и пакеты, направленные RP-интерфейсу. Несмотря на то что эти операции являются обычными, они могут использоваться в качестве атаки типа "отказ в обслуживании" (denial-of-service attack — DoS) против процессора маршрутизации.

С помощью параметра *pps* (количество пакетов в секунду) можно ограничить скорость отправляемых RP процессору пакетов (от 0 до 1 000 000 пакетов в секунду, стандартное значение равно нулю или без ограничения).

Отображение информации о CEF-коммутации

Для отображения полезных сведений по CEF-коммутации можно использовать команды коммутатора, приведенные в табл. 8.3. Ниже представлены обозначения, позволяющие отличать команды для различных платформ коммутации:

- **COS**. Операционная система Catalyst, используемая в блоке Supervisor коммутатора Catalyst 6000.
- **IOS**. Программное обеспечение операционной системы Cisco IOS, которое используется в плате MSFC2 коммутатора Catalyst 6000, коммутаторах Catalyst 2948G-L3, 4908G-L3, 4000 Supervisor III и 3550.
- **Sup IOS**. Операционная система Supervisor IOS. Программное обеспечение Cisco IOS для блока Supervisor 2 коммутатора Catalyst 6000.

Совет

Следует помнить, что в коммутаторе Catalyst 6000 CEF-функция разделена между платами MSFC2 и PFC2. Если используются команды для вывода информации о CEF на плате MSFC2, то отображается только часть информации о создании базы данных FIB и таблиц смежности. Хотя плата MSFC2 может использовать собственную CEF-функцию для передачи пакетов, не коммутируемых на третьем уровне платой PFC2, как правило, плата MSFC2 только создает, загружает и обновляет базу FIB и таблицы смежности для платы PFC2.

Для отображения информации о характеристиках коммутации третьего уровня необходимо ввести команды на том модуле Catalyst Supervisor, где расположена плата PFC2.

Очевидно, если имеется коммутатор, работающий с IOS-кодом, то будет отображен один блок интегрированной CEF-информации, которая создается на плате MSFC2 и используется модулем PFC2.

Таблица 8.3. Команды для отображения CEF-информации

Функция отображения	Операционная система коммутатора	Команда
База данных FIB, созданная платой MSFC2 или RP-процессором	COS	Нет

	IOS	`show ip cef [[unresolved [detail]] | [detail | summary]]`
		`show ip cef [network [mask]] [longer-prefixes] [detail]`
		`show ip cef [vlan number] [detail]`

	Sup IOS	`show mls cef [prefix] [mask]` `show mls cef [module number | summary]`

База FIB, используемая платой PFC2	COS	`show mls cef`
		`show mls entry cef ip [[ip-addr/[mask-len]]`
		`show mls entry cef ipx [[ipx-addr/[mask-len]]`

	IOS	Нет

	Sup IOS	`show mls cef [module number | summary]`
		`show mls cef ip [[prefix [mask | module number]] | [module number]]`
		`show mls cef ipx [[prefix [mask | module number]] | [module number]]`

Таблица смежности	COS	`show mls entry cef adjacency`
		`show mls entry cef ip [[next-hop-addr/[32]] adjacency`
		`show mls entry cef ipx [[next-hop-addr/[mask len]] adjacency`

	IOS	`show adjacency [type number] [detail] [summary]`
		`show ip cef adjacency type number ip-prefix [detail]`
		`show ip cef adjacency [discard | drop | glean | null | punt] [detail ]`

Функция отображения	Операционная система коммутатора	Команда
	Sup IOS	`show mls cef adjacency [count | mac-address number] [module number]`
Мультадресатные записи CEF	COS	`show mls multicast entry [all] [short | long]` `show mls multicast entry [mod] [vlan vlan-id] [group ip-addr] [source ip-addr] [long | short]`
	IOS	`show mls ip multicast group group-address [interface type number | statistics]`
	Sup IOS	`show mls cef ip multicast [{prefix [mask | module num]}]]`
Активные VLAN-интерфейсы, используемые платой MSFC2 для CEF	COS	`show mls cef [interface [vlan]` `show mls cef mac`
	IOS	Нет
	Sup IOS	`show mls cef mac`

При отображении содержимого таблицы FIB каждая запись выводится с полем "FIB-типа". Значения этого поля описаны ниже.

- *Прием (receive).* Получатель связан с MSFC-интерфейсом (маска длиной 32).
- *Подключен (connected).* Получатель связан с подключенной сетью.
- *Распознан (resolved).* Получатель связан с достоверным адресом и смежностью следующего транзитного перехода.
- *Уничтожение пакетов (drop).* Связанные с данным получателем пакеты отбрасываются.
- *Инвертированная маска (wildcard).* Запись, соответствующая всем отброшенным или перенаправленным MSFC в случае, если стандартный маршрут отсутствует.
- *Стандартный маршрут (default).* Инвертированная маска указывает на стандартный маршрут.

При отображении таблицы смежности каждая запись выводится с полем типа смежности (AdjType).

- *drop, null, loopbk.* Отбрасываемые пакеты не перенаправляются.
- *frc drp.* Удаленные смежности ввиду подавления запросов ARP.
- *punt.* Перенаправление на плату MSFC для дальнейшей обработки.
- *no r/w.* Перенаправление ввиду неполной перезаписи пакета.

8.3: экспорт данных NetFlow

- Статистика по графику коммутации третьего уровня может быть собрана и отправлена внешнему приложению для накопления и анализа. Эта функция осуществляется с помощью *экспорта данных NetFlow (NetFlow Data Export — NDE)*.
- Коммутаторы, использующие функцию MLS для коммутации третьего уровня, могут отправлять данные об устаревших потоках с помощью функции NDE, что является естественным расширением MLS-коммутации, поскольку коммутатор использует данные кэша потоков.
- Коммутаторам, использующим CEF-функцию, не свойственно использование кэша потоков и, следовательно, они неспособны предоставлять статистику посредством службы NDE. Однако коммутатор Catalyst 6000 PFC2/MSFC2 поддерживает независимый от CEF-процесса NetFlow-кэш строго для экспорта данных с помощью функции NDE.
- Данные NetFlow могут отправляться в нескольких версиях:
  - **NDE версии 1**. Используется в традиционных системах, записи включают в себя специфическую информацию о потоке IP-графика и интерфейсах, которые используются для его передачи.
  - **NDE версии 5**. Добавляется номер последовательности для предотвращения потерь дейтаграмм, а также номер автономной системы BGP *(Border Gateway Protocol BGP autonomous system — автономная система протокола пограничного шлюза)* для потока данных.
  - **NDE версии 7**. Используется для создания отчетов по данным, полученным от коммутаторов Catalyst. В коммутаторе Catalyst 6000 MSFC эта версия не поддерживается.
  - **NDE версии 8**. Используется для создания отчетов по суммарным потокам данных от маршрутизаторов, коммутаторы Catalyst 5000 с платами NFFC и коммутаторы Catalyst 6000, использующих MLS- или CEF-коммутацию. Эта версия не поддерживается платой Catalyst 6000 MSFC.
- Функция NDE осуществляет экспорт статистики согласно маске MLS-потока, используемой коммутатором. Чтобы получить подробные записи потоков, необходимо использовать маску "полного" потока.

Конфигурирование функции NDE

1. Запуск функции NDE на процессоре маршрутизации.

   В первую очередь необходимо сконфигурировать MLS-коммутацию на RP-процессоре. Более подробные указания приведены в разделе "8.1. многоуровневая коммутация" (этап 1).

2. Запуск функции NDE в блоке коммутации.

   a) Идентификация коллектора данных потока (flow data collector).

Система COS	`set mls nde collector udp port`
Система IOS	`ip flow-export destination collector udp-port` (режим глобальной конфигурации)

Узел, на котором выполняется приложение-коллектор, идентифицируется с помощью параметра `collector` (IP-адрес или имя). В дополнение к этому необходимо задать NDE UDP-порт (*udp port*), соответствующий номеру порта, который используется приложением-коллектором.

б) Идентификация NDE-источника

Система COS	Нет
Система IOS	`ip flow-export source {interface interface-number} {null 0} | {port channel num ber} {vlan vlan-id}` (режим глобальной конфигурации)

NDE-пакеты получают IP-адрес отправителя от указанного интерфейса. Для COS-коммутаторов в качестве адреса отправителя используется адрес административного интерфейса sc0. В качестве источника всегда следует использовать интерфейс обратной петли (loopback interface), поскольку он постоянно находится в активном состоянии и всегда доступен. Можно использовать интерфейс[2] null 0 и в том случае, если адрес отправителя NDE-информации не требуется включать в состав экспортируемых данных.

в) Включение функции NDE.

Система COS	`set mls nde version {1 | 7 . 8}` `set mls nde {enable | disable}`

Система IOS	`mls nde sender [version version]`
	или
	`ip flow-export version {1 {5 [origin-as | peer-as]} , {6 [origin-as | peer-as]}}`
	(обе команды вводятся в режиме глобальной конфигурации)

Можно указывать 1, 7 (стандартную) или 8 версии (*version*) NDE. На IOS-коммутаторе команда `mls nde sender` устанавливает версию NDE, которая используется платой PFC2 устройства Catalyst 6000 (собственной или Supervisor IOS), тогда как команда `ip flow-export` настраивает NDE-версию для маршрутизируемых потоков на платах MSFC/MSFC2 (MSFC IOS).

3. Фильтрация экспортируемых данных.

Система COS	`set mls nde flow {include | exclude} [destination` *ip-addr spec*`] [source` *ip addr-spec*`] [protocol` *protocol*`] [src-port` *src-port*`] [dst- port` *dst-port*`]`
Система IOS	`mls nde flow { | exclude} {{dest-port` *port-num*`} , {destination` *ip-addr ip-mask*`}} | {protocol {tcp , udp}} ! {source` *ip-addr ip mask*`} , {src-port` *port-num*`}}` (режим глобальной конфигурации)

[2] *Несуществующий логический интерфейс. — Прим. перев.*

Потоки данных можно экспортировать, только если указано ключевое слово include и критерии соответствия. Если используется ключевое слово exclude, потоки к узлу не будут фиксироваться. Указанные ключевые слова являются взаимоисключающими, поскольку в одной команде может использоваться только один такой фильтр. Однако можно использовать несколько команд, для того чтобы построить оба фильтра — include и exclude.

Потоки могут отбираться на основании адреса получателя (destination), отправителя (source), порта получателя (dst-port, 0 соответствует любому значению), порта отправителя (src-port, 0 соответствует любому значению) и протокола (IOS: tcp или udp; COS: от 0 до 255 или ip, ipinip, icmp, igmp, tcp или udp, 0 соответствует любому значению). Для IOS-коммутаторов адреса указываются в виде *ip-addr ip-mask* (адрес и маска). Для COS-коммутаторов адреса можно задавать в одном из трех вариантов: *ip-addr*, *ip-addr/ip-mask* или *ip-addr/maskbits*.

Следует заметить, что IOS-команда допускает использование только одного критерия, тогда как COS-команда позволяет использовать любую комбинацию параметров.

Пример конфигурирования функции NDE

NDE конфигурируется на коммутаторе с интегрированными модулями RP и SE. Коллектор (NetFlow Collector) размещается по адресу с адресом 192.168.177.10 и для NDE-обмена использует UDP-порт номер 5000. Коммутатор отправляет NDE-данные, используя адрес отправителя 192.168.40.1, полученный от административного интерфейса sc0 (в системе COS) или VLAN-интерфейса 900 (в системе IOS).

При отправке NDE данных NetFlow-коллектору коммутатор включает только информацию по потокам трафика для TCP-порта номер 80.

Система COS	`set nls nde 192.168.177.10 5000`
	`set nls nde version 7`
	`set nls nde enable`
	`set nls nde flow include protocol tcp dst-port 80`
	`set interface sc0 900 192.168.40.1 255.255.255.0`

Система IOS	`ip flow-export destination 192.168.177.10 5000`
	`ip flow-export source vlan 900`
	`nls nde sender version 7`
	`nls nde flow include dest port 80`
	`interface vlan 900`
	(все указанные выше команды вводятся в режиме глобальной конфигурации)
	`ip address 192.168.40.1 255.255.255.0`
	(в режиме конфигурирования интерфейса)

Отображение информации о функции NDE

Для отображения полезной информации о функции NDE можно использовать команды, приведенные в табл. 8.4.

Таблица 8.4. Команды для отображения NDE-информации

Функция отображения	Операционная система коммутатора	Команда
Версия и активность NDE	COS	`show mls nde`
	IOS	`show mls netflow` (режим привилегированного или непривилегированного пользователя)

8.4: резервирование модулей MSFC в одном устройстве

- Резервные блоки Supervisor с запасными процессорами маршрутизации (MSFC/MSFC2) поддерживаются только на коммутаторах Catalyst серии 6000.
- Только одна плата MSFC является выделенной (designated) и активной. Другая плата загружается, синхронизирует свою конфигурацию с конфигурацией активной платы MSFC, запускает процессы какого-либо протокола маршрутизации, но удерживает все интерфейсы в состоянии "линия отключена" (line down), поэтому обмен сетевым трафиком не происходит.
- Только выделенная и активная MSFC-плата занимается созданием записей кэша MLS-потоков для блока коммутации, а также в создании и загрузке CEF FIB-базы и таблиц смежности для блока коммутации PFC2.
- При сбое *режим с единственным маршрутизатором (Single Router Mode — SRM)* приводит к тому, что невыделенная плата MSFC включает свои интерфейсы и предоставляет возможность конвергенции для используемых протоколов маршрутизации. В течение этого времени выделенная PFC-плата блока Supervisor продолжает использовать существующую информацию по коммутации третьего уровня, предоставленную вышедшей из строя платой MSFC, до тех пор, пока новая плата MSFC не обеспечит обновления.
- Оба MSFC-модуля должны использовать один и тот же образ программного обеспечения операционной системы Cisco IOS — версию 12.1(8a)E2 или более позднюю.
- Модули Supervisor должны функционировать в режиме высокой надежности для поддержки функции SRM с помощью программного обеспечения блока Supervisor (Supervisor Engine Software) версии 6.3(1) или более поздней.
- В режиме SRM использование протокола HSRP *(Hot Standby Router Protocol — резервный протокол маршрутизации)* не является необходимым, поскольку в один момент времени в шасси активна только одна плата MSFC. Этот протокол следует использовать в том случае, если имеются другие MSFC-устройства или устройства третьего уровня, присутствующие в тех же VLAN-сетях, что и SRM-службы MSFC. Такие устройства впоследствии могут использоваться в качестве резервных шлюзов в этих сетях.

Конфигурирование функции

1. Включение режима высокой надежности блока Supervisor.

Система COS	`set system highavailability enable`
Система IOS	Нет

Высокая надежность требуется на резервных блоках Supervisor для того, чтобы во время сбоя Supervisor или платы MSFC соответствующим образом поддерживалась информация коммутации третьего уровня. Эта информация используется во время сбоя платы MSFC, пока переключенная плата ожидает конвергенции своей маршрутной информации.

2. Включение режима SRM на выделенной плате MSFC.

Система COS	Нет

Система IOS	`redundancy`
	(режим глобальной конфигурации)
	`high-availability`
	(режим конфигурирования резервирования)
	`single-router-mode`
	(режим конфигурирования службы высокой надежности)

Плата, в которой SRM-режим сконфигурирован раньше, становится выделенной. Режим SRM не активируется до тех пор, пока не будет включен на обеих платах.

3. Регулировка задержки конвергенции при восстановлении после сбоя (*необязательно*)

Система COS	Нет
Система IOS	`single-router-mode failover table-update-delay` *seconds* (режим конфигурирования службы высокой надежности)

Если переключенная плата MSFC обнаруживает сбои и становится активной, то прежде чем отправлять какую-либо информацию по коммутации третьего уровня SE-блоку (т.е. модулю PFC/PFC2), она ожидает в течение определенного времени (*seconds*, стандартно 120 секунд). Чтобы обеспечить соответствующую конвергенцию каких-либо сконфигурированных протоколов маршрутизации, задержку можно настроить более точно.

4. Включение SRM-режима на невыделенной MSFC-плате

Система COS	Нет

Система IOS	`redundancy`
	(режим глобальной конфигурации)
	`high-availability`
	(режим конфигурирования резервирования)
	`single-router-mode`
	(режим конфигурирования службы высокой надежности)

После ввода указанных команд невыделенная плата MSFC принимает конфигурацию от выделенной MSFC.

5. Сохранение действующей конфигурации на невыделенной плате MSFC.

Система COS	Нет
Система IOS	`copy running-config startup-config`

Конфигурация выделенной MSFC-платы автоматически сохраняется на невыделенной плате.

6. Перезагрузка невыделенной платы MSFC.

Система IOS	`reload` (режим привилегированного пользователя)

После повторной загрузки невыделенная плата MSFC входит в режим SRM, при этом ее интерфейсы находятся в состоянии "линия неактивна".

Отображение информации, касающейся режима SRM

В табл. 8.5 приведены команды коммутатора, которые можно использовать для отображения полезной информации по SRM-избыточности.

Таблица 8.5. Команды для отображения SRM-информации

Функция отображения	Операционная система коммутатора	Команда
SRM-состояние	COS	Нет
	IOS	`show redundancy` (режим привилегированного или непривилегированного пользователя)

8.5: MSFC-резервирование с синхронизацией конфигурации

- В режиме синхронизированной конфигурации (config sync mode) обе платы MSFC постоянно активны. Все интерфейсы и процессы маршрутизации доступны и активны на обоих модулях.
- Одна из MSFC-плат является "выделенной" и поддерживает главные копии начальной и действующей конфигурации. Другая плата является "невыделенной" и получает конфигурацию от выделенного модуля.
- Состояние невыделенной платы MSFC можно отслеживать посредством EXEC-сеанса, однако плата не предоставляет доступ к режиму конфигурации.
- Режим синхронизированной конфигурации допускает немедленное восстановление после сбоев, поскольку обе MSFC-платы всегда активны. Все VLAN-интерфейсы, а также все сконфигурированные протоколы маршрутизации активны на обоих модулях.

- В создании записи кэша MLS-потока может быть задействован любой из двух избыточных MSFC-модулей. Плата PFC обладает информацией об обоих RP-модулях.
- При CEF-коммутации только выделенный MSFC-модуль загружает FIB-базу и таблицы смежности в другие модули. Выделенным является тот модуль MSFC, который инициализируется первым в модульном гнезде с наименьшим номером.
- Для предоставления адреса резервного шлюза в каждой VLAN-сети следует использовать протокол HSRP. Обе MSFC-платы совместно используют общий адрес шлюза, оперируя при этом собственными уникальными адресами интерфейсов. Более подробная информация по протоколу HSRP приведена в разделе "Резервирование маршрутизаторов с помощью протокола HSRP".
- После включения и активизации режима синхронизированной конфигурации любые конфигурационные изменения, сделанные на выделенной MSFC-плате, автоматически синхронизируются с невыделенным модулем.
  - Каждый раз при вводе команд `write mem` и `copy source startup-config` обновляется начальная конфигурация MSFC-модулей.
  - Каждый раз при вводе команды `copy source running-config` обновляется действующая конфигурация MSFC-модулей.
  - По мере ввода команд в режиме конфигурации они также отправляются и выполняются на невыделенном MSFC-модуле.

Настройка резервирования с синхронизированной конфигурацией

1. Включение режима `config-sync` на выделенном модуле MSFC

Система COS	Нет

Система IOS	`redundancy`
	(режим глобальной конфигурации)
	`high-availability`
	(режим конфигурирования резервирования)
	`config-sync`
	(режим конфигурирования службы высокой надежности)

Выделенный модуль можно выбрать произвольно. Он является первым модулем, который инициализируется с включенной синхронизацией конфигурации. Как правило, проще начать с MSFC-платы, расположенной в гнезде 1 шасси (слот 15).

Совет

С этого момента режим `config-sync` административно включен. Однако для него требуется, чтобы выделенная плата обладала конфигурационной информацией для обеих MSFC-плат, поскольку она поддерживает главную конфигурацию. Поэтому в некоторые команды необходимо включить ключевое слово `alt`, которое позволяет указать параметры для невыделенной платы MSFC.

Описываемый режим не будет активным до тех пор, пока не будет задана альтернативная информация и в невыделенной плате MSFC не будет включен режим синхронизации конфигурации.

2. Конфигурирование альтернативных параметров на выделенном модуле MSFC.

Совет

Если для некоторых функций уже имеется выделенный модуль MSFC, то чтобы установить альтернативные значения для невыделенного модуля, придется вновь ввести некоторые команды. Следует помнить о том, что выделенный модуль поддерживает главную конфигурацию для обеих плат MSFC. Чтобы задать в одной командной строке значения для обеих плат (MSFC в гнезде 1 до ключевого слова `alt` и MSFC в гнезде 2 после него), синтаксис некоторых команд нужно изменить.

Если в качестве выделенного избран модуль в гнезде 1, то параметры команды естественно расположить в виде "выделенный модуль `alt` невыделенный модуль".

В большинстве случаев после ключевого слова `alt` повторяется вся команда, а не только ее параметры.

а) Имя узла MSFC (*необязательно*).

Система COS	Нет
Система IOS	`hostname` *hostname* `alt hostname` *hostname* (режим глобальной конфигурации)

б) Стандартный шлюз (*необязательно*).

Система COS	Нет
Система IOS	`ip default-gateway` *ip-address* `alt ip default-gateway` *ip-address* (режим глобальной конфигурации)

в) Идентификатор BGP-маршрутизатора (*необязательно*).

Система COS	Нет
Система IOS	`router bgp` *as-number* (режим глобальной конфигурации) `bgp router-id` *ip-address* [`alt` *ip-address*] (режим конфигурирования маршрутизатора)

г) Идентификатор OSPF-маршрутизатора (*необязательно*).

Система COS	Нет
Система IOS	`router ospf` *process id* (режим глобальной конфигурации) `router-id` *ip-address* [`alt` *ip-address*] (режим конфигурирования маршрутизатора)

д) IP-адрес интерфейса (*необязательно*).

Система COS	Нет
Система IOS	`ip address` *ip-address mask* [`secondary`] `alt` [`no`] `ip address` *ip-address mask* [`secondary`] (режим конфигурирования интерфейса)

Команду необходимо повторить для каждого интерфейса. Если плата MSFC в гнезде 7 не имеет соответствующего активного интерфейса, используется ключевое слово no.

Совет

Следует уделить особое внимание вводу команды для дополнительных (secondary) адресов. Как правило, пользователи помнят об использовании ключевого слова **secondary** в части команды для выделенного модуля и весьма часто забывают указать его для альтернативной части. Если ключевое слово **secondary** не указано, то введенный IP-адрес заменяет любой ранее заданный главный адрес.

c) Номер IP-адреса HSRP группы (*необязательно*).

Система COS	Нет
Система IOS	**standby** [*group*] **ip** [*ip-address* [**secondary**]] **alt** [**no**] **standby** [*group*] **ip** [*ip-address* [**secondary**]] (режим конфигурирования интерфейса)

Как и для описанной выше команды, при использовании дополнительных адресов следует убедиться, что для обоих MSFC-интерфейсов введено ключевое слово **secondary**.

ж) HSRP-приоритет и задержка (*необязательно*).

Система COS	Нет
Система IOS	**standby** [*group*] **priority** *priority* [**preempt** [**delay** *delay*]] **alt standby** [*group*] **priority** *priority* [**preempt** [**delay** *delay*]] (режим конфигурирования интерфейса)

Для двух MSFC-интерфейсов следует задавать различные значения HSRP приоритета (от 1 до 255, стандартное значение равно 100). Большее значение рекомендуется выбирать для интерфейса, который станет активным HSRP-адресом.

з) IPX-сеть (*необязательно*).

Система COS	Нет
Система IOS	**ipx network** *network* [**encapsulation** *encap-type* [**secondary**]] **alt ipx network** *network* [**encapsulation** *encap-type* [**secondary**]] (режим конфигурирования интерфейса)

3. Сохранение конфигурации на выделенном MSFC модуле.

Система COS	Нет
Система IOS	**copy running-config startup-config** (режим привилегированного пользователя)

4. Включение режима синхронизации конфигурации (**config-sync**) на невыделенном MSFC-модуле.

Система COS	Нет

Система IOS	`redundancy`
	(режим глобальной конфигурации)
	`high-availability`
	(режим конфигурирования резервирования)
	`config-sync`
	(режим конфигурирования службы высокой надежности)

Режим `config-sync` не будет функционировать до тех пор, пока он не будет настроен и включен на обоих MSFC-модулях. Сразу после включения этого режима на обоих модулях запускается одноминутный таймер, который позволяет им стабилизироваться и обменяться информацией. По истечении этого времени выделенный модуль копирует свою действующую конфигурацию на вычисленный модуль MSFC.

Отображение информации о config-sync-резервировании

Приведенные в табл. 8.6 команды коммутатора можно использовать для отображения текущей информации, касающейся избыточности с синхронной конфигурацией.

Таблица 8.6. Команды отображения состояния для избыточности с синхронной конфигурацией

Функция отображения	Операционная система коммутатора	Команда
Config-sync-состояние	COS	Нет
	IOS	`show redundancy` (режим привилегированного или непривилегированного пользователя)

8.6: резервирование маршрутизаторов с помощью протокола HSRP

- Процессоры маршрутизации на одном или разных шасси могут совместно использовать адреса избыточных однотипных VLAN-сетей путем применения *резервного протокола маршрутизации (Hot Standby Router Protocol — HSRP)*.
- Процессоры маршрутизации, совместно использующие общий HSRP IP-адрес, должны принадлежать одной HSRP группе.
- HSRP-адрес появляется в сети со специальным виртуальным MAC-адресом 00-00-0C-07-AC-XX, где XX — номер HSRP-группы (от 0 до 255). Узлы в HSRP VLAN используют этот MAC-адрес в качестве стандартного шлюза.
- Несмотря на то что на интерфейсе используется протокол HSRP, каждый процессор маршрутизации продолжает поддерживать на VLAN-интерфейсе собственные уникальные IP- и MAC-адреса, которые используются другими маршрутизаторами для трафика протокола маршрутизации.

- При включении HSRP-группы HSRP-устройство с наименьшим приоритетом становится активным маршрутизатором, тогда как устройство со вторым по величине приоритетом остается в ждущем режиме. Все остальные HSRP-устройства в этой группе остаются в состоянии "прослушивания", ожидая сбоя активного устройства. Выбор нового активного маршрутизатора происходит только при выходе из строя активного устройства. Активный ранее маршрутизатор (имеющий наивысший приоритет) может возобновить свою активную роль путем приоритетного прерывания обслуживания (preempting) других HSRP-маршрутизаторов группы.
- HSRP-устройства обмениваются информацией, отправляя hello-сообщения по протоколу UDP на многоадресатный адрес 224.0.0.2. Такие сообщения по умолчанию отправляются через каждые 3 секунды.
- Устройства VLAN-сети используют HSRP-адрес в качестве стандартного шлюза. Если одно из HSRP-устройств выходит из строя, всегда находится другое устройство, принимающее на себя функции стандартного шлюза на данном адресе.

Конфигурирование функции

1. Назначение номера и адреса HSRP-группы

Система COS	Нет
Системы IOS	`standby [group-number] ip [ip-address [secondary]]` (режим конфигурирования интерфейса)

VLAN-интерфейс входит в состав номера HSRP-группы *group-number* (от 0 до 255, стандартно — 0) как HSRP IP-адрес *ip-address*. Если этот адрес соответствует дополнительному адресу на фактическом VLAN-интерфейсе, следует использовать ключевое слово `secondary`, в результате чего появляется возможность активизировать HSRP-адреса как для главного, так и для дополнительного интерфейсов.

Номер группы и IP-адрес должны быть одинаковыми на всех устройствах третьего уровня, участвующих в HSRP в данной VLAN-сети. В результате виртуальный MAC-адрес также является идентичным для всех HSRP-устройств.

Совет

Обычной практикой является использование номера VLAN-сети в качестве номера HSRP-группы для удобства ссылок. Однако комбинация устройства Catalyst 6000 и платы PFC2 или MSFC2 поддерживает не более 16 различных HSRP-групп (с номерами от 1 до 255). Вместе с тем можно повторно задействовать один номер группы на нескольких VLAN-интерфейсах при условии, что между VLAN-сетями не существует мостов.

2. Установка HSRP-приоритета (*необязательно*).

Система COS	Нет
Системы IOS	`standby [group-number] priority priority [preempt [delay minimum delay]]` (режим конфигурирования интерфейса)

Чтобы стать активным устройством, интерфейс согласовывает параметры с другими HSRP-устройствами группы. Следует назначить приоритет (*priority* от 1 до 255, стандартно — 100) для каждого HSRP-устройства так, чтобы активным стало устройство с наивысшим приоритетом (наивысший приоритет равен 255). Чтобы добиться ожидаемого результата выбора в случае сбоя активного устройства, следует отрегулировать приоритеты всех остальных устройств.

Если в активном устройстве с наивысшим приоритетом произошел сбой, то прежде чем снова стать активным, это устройство ожидает сбоя в новом активном устройстве с меньшим приоритетом. Чтобы разрешить устройству немедленно выполнить активную роль, используется ключевое слово **preempt**. Можно добавить ключевые слова **delay minimum**, чтобы задержать приоритетное прерывание обслуживания на определенное время (*delay*, от 0 до 3600 секунд, стандартно — 0 или задержка отсутствует) после перезапуска коммутатора третьего уровня, в результате чего пройдет некоторый период времени для конвергенции протокола маршрутизации.

3. Использование HSRP-аутентификации (*необязательно*).

Система COS	Нет
Система IOS	**standby** [*group-number*] **authentication** *string* (режим конфигурирования интерфейса)

Стандартно любое устройство может принимать участие в HSRP-процессе. Чтобы устройства аутентифицировали друг друга с помощью строки *string* (текстовая строка длиной до восьми символов), в качестве простого текстового ключа используется ключевое слово **authentication**.

4. Тонкая настройка HSRP-таймеров (*необязательно*).

Система COS	Нет
Система IOS	**standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime* (режим конфигурирования интерфейса)

С помощью параметра *hellotime* можно задать период времени между отправкой hello-сообщений (от 1 до 254 секунд, стандартно — 3 секунды, или от 50 до 999 миллисекунд), используя ключевое слово **msec**.

HSRP-устройства ожидают hello-сообщений от активного устройства до истечения времени удержания, после чего активное устройство объявляется отключенным, а его функции принимает на себя следующее устройство с наиболее высоким приоритетом. Время удержания (*holdtime*) можно задать с помощью ключевого слова **msec** (до 255 секунд, стандартно — 10 секунд, или до 3000 миллисекунд). Следует убедиться, что время удержания на всех HSRP-устройствах группы согласовано.

Совет

Чтобы получать уведомление о смене активного устройства, можно включить SNMP-прерывание из базы MIB технологии HSRP. Для этого используется команда **snmp-server enable traps hsrp**. Более подробная информация по SNMP-конфигурации приводится в разделе "12.2: простой протокол управления сетью".

Пример конфигурирования протокола HSRP

Два коммутатора третьего уровня имеют интерфейсы в сети VLAN 199. В качестве этих устройств могут использоваться два MSFC-модуля в одном шасси Catalyst 6000 или в двух отдельных шасси либо два коммутатора Catalyst 3550, и т.д.

В нашем примере мы используем HSRP-группу 1. В действительности HSRP-группа 1 может использоваться на каждом VLAN-интерфейсе при условии, что сконфигурированные мосты второго уровня не существуют. HSRP-устройства совместно используют IP-адрес 192.168.104.1, поэтому для узлов VLAN-сети 199 всегда существует доступный стандартный шлюз. Следует заметить, что IP-адрес 192.168.104.1 представляется как виртуальный MAC-адрес 00-00-0C-07-AC-01 (01 указывает на HSRP-группу 1).

Hello-таймеры устройств настроены на 3 секунды, а время удержания равно 40 секундам. Устройству А назначается приоритет 210, в результате чего у него появляется преимущество перед устройством Б, приоритет которого равен 200, при выборе активного узла. Устройство А настраивается на приоритетное прерывание обслуживания всех остальных низкоприоритетных устройств, которые могут стать активными, но только по истечении не менее 60 секунд после его перезапуска. Такие настройки позволят устройству при необходимости взять на себя роль активного. Приоритетное прерывание обслуживания (preempt) не является необходимым в ситуации с двумя HSRP-маршрутизаторами, поскольку два устройства всегда будут выбирать компромиссное решение в отношении активной роли. Приоритетное прерывание может оказаться полезным в ситуации, когда в состав группы входят более двух HSRP-устройств.

Наконец, HSRP-устройства используют в ходе HSRP-связи строку *myhsrpkey* в качестве простой формы аутентификации. Если какой-либо узел попытается использовать HSRP-сообщение без ключа аутентификации, остальные устройства не будут его рассматривать.

Конфигурация третьего уровня для устройства А.

Система COS	Нет

Система IOS	`interface vlan 199`
	(режим глобальной конфигурации, остальные команды вводятся в режиме конфигурирования интерфейса)
	`standby 1 ip 192.168.104.1`
	`standby 1 priority 210 preempt delay 60`
	`standby 1 authentication myhsrpkey`
	`standby 1 timers 3 40`

Конфигурация третьего уровня для устройства Б.

Система COS	Нет

Система IOS	`interface vlan 199`
	(режим глобальной конфигурации, остальные команды вводятся в режиме конфигурирования интерфейса)
	`standby 1 ip 192.168.104.1`
	`standby 1 priority 200 preempt`
	`standby 1 authentication myhsrpkey`
	`standby 1 timers 3 40`

Отображение сведений по протоколу HSRP

Для отображения полезной информации о работе протокола **HSRP** на интерфейсах можно использовать команды коммутатора, приведенные в табл. 8.7.

Таблица 8.7. Команды для отображения HSRP-информации

Функция отображения	Операционная система коммутатора	Команда
Краткие сведения о HSRP-состоянии	COS	Нет
	IOS	`show standby brief`
Протокол HSRP на определенном VLAN-интерфейсе	COS	Нет
	IOS	`show standby vlan` *vlan-number* [*hsrp-group*] `[brief]`

Дополнительная литература

Рекомендуемые источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Технология MLS

Кеннеди Кларк, Кевин Гамильтон. *Практика коммутации в локальных сетях Cisco*. ИД "Вильямс", 2003.

Tim Boyles and David Hucaby. *CCNP Switching Exam Certification Guide*. Cisco Press

CEF-коммутация

Вилжей Болдиграгда, Кэртис Мэрфи, Расс Уайт. *Структура операционной системы Cisco IOS*. ИД "Вильямс", 2002.

Поиск и устранение неисправностей в однонаправленной IP-коммутации CEF на коммутаторах Catalyst 6000 с модулем Supervisor 2 в гибридном режиме (How-To Troubleshoot Unicast IP Routing CEF on Catalyst 6000s with a Supervisor 2 in Hybrid Mode), www.cisco.com/warp/public/473/128.html.

Функция экспорта данных NetFlow

Cisco IOS NetFlow: www.cisco.com/warp/public/732/Tech/netflow/.

Справочник по решениям на основе служб NetFlow (NetFlow Services Solutions Guide), www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/netsol/nfwhite.htm.

Резервирование маршрутизаторов

Проектирование кампусных иерархических сетей — принципы и структура (Cisco Campus Network Design — Principles and Architecture). www.cisco.com/warp/public/cc/so/neso/lnso/cpso/gcnd_wp.htm.

Причины возникновения, поиск и устранение проблем HSRP в сетях на основе коммутаторов корпорации Cisco (Understanding and Troubleshooting HSRP Problems in Catalyst Switch Networks). www.cisco.com/warp/public/473/62.shtml.

Использование протокола HSRP для обеспечения отказоустойчивой IP-маршрутизации (Using HSRP for Fault-Tolerant IP Routing). www.cisco.com/univercd/cc/td/doc/cisintwk/ics/cs009.htm

В этой главе...

- 9.1: **IGMP-прослушивание**. В этом разделе описывается конфигурирование коммутатора для ограничения многоадресного трафика путем прослушивания сообщений межсетевого протокола управления группами (*Internet Group Management Protocol — IGMP*).
- 9.2: протокол **CGMP**. В разделе описаны этапы конфигурирования, необходимые для предоставления коммутатору и маршрутизатору возможности управлять многоадресным трафиком путем обмена сообщениями *протокола управления группами Cisco* (*Cisco Group Management Protocol — CGMP*).
- 9.3: протокол **GMRP**. В разделе описываются этапы конфигурирования, необходимые для настройки коммутатора на использование протокола *GARP-регистрации многоадресного маршрутизатора* (*GARP Multicast Router Protocol — GMRP*) для информационного обмена с многоадресными узлами.
- 9.4: протокол **RGMP**. В разделе описан процесс использования коммутатором и маршрутизатором протокола *групповых управляющих портами маршрутизатора* (*Router-port Group Management Protocol — RGMP*) для сдерживания многоадресного трафика на маршрутизаторах в сети.

Глава 9

Многоадресатные службы

Совет

Многоадресатные функции в сети рекомендуется выбирать на основании типа используемых коммутаторов. Обычно требуется многоадресатная IP-маршрутизация, в которой используется *независимое от протокола многоадресатное вещание (Protocol Independent Multicast — PIM)*. Маршрутизация такого типа обычно конфигурируется только на маршрутизаторе или платформах третьего уровня. Другие многоадресатные функции рекомендуется конфигурировать на основании данных, приведенных в табл. 9.1, и ряда правил, описанных ниже.

- IGMP-прослушивание используется почти на всех моделях коммутаторов начального уровня. IGMP-прослушивание не требует какого-либо дополнительного взаимодействия с маршрутизаторами.
- Протокол CGMP используется для моделей коммутаторов начального уровня, неспособных использовать IGMP-прослушивание. Необходимо настраивать протокол CGMP на любых непосредственно подключенных интерфейсах маршрутизатора так, чтобы CGMP-информация передавалась коммутаторам.
- Протокол CGMP следует использовать для контроля многоадресатного трафика, только если имеющиеся многоадресатные узлы способны использовать оба протокола IGMP и CGMP. Платформа коммутации также должна поддерживать протокол CGMP.
- Протокол RGMP используется для отсечения многоадресатного трафика к маршрутизаторам и, более того, для исключения портов, связанных с маршрутизаторами, которым не требуются многоадресатные группы. Протокол RGMP должны поддерживать как маршрутизаторы, так и коммутаторы.

Таблица 9.1. Многоадресатные функции коммутаторов

Модель коммутатора Catalyst	PIM-маршрутизация	IGMP-прослушивание	Протокол CGMP	Протокол GMRP	Протокол RGMP
2900XL	Нет	Нет	Есть	Нет	Нет
3500XL	Нет	Нет	Есть	Нет	Нет
3550	Есть	Есть	Нет	Нет	Нет
5000	Нет	Есть	Есть	Есть	Есть
4000 Sup I, II	Нет	Есть	Есть	Есть	Нет

Окончание табл. 9.1

Модель коммутатора Catalyst	PIM-маршрутизация	IGMP-прослушивание	Протокол CGMP	Протокол GMRP	Протокол RGMP
4000 Sup III	Есть	Есть	Есть	Нет	Нет
5000 с платой PFC	Нет	Есть	Нет	Нет	Нет
6000 с платами PFC/MSFC	Есть	Есть	Нет	Есть (COS)	Есть
6000 с платами PFC2/MSFC2	Есть	Есть	Нет	Есть (COS)	Есть

Адресация в технологии многоадресатной передачи

- Многоадресатные IP-потоки могут обозначаться двумя способами.
  - (S, G) — уникальная древовидная структура кратчайшего пути между отправителем и многоадресатными получателями, читается как "S запятая G". S — адрес одноадресатного IP-отправителя, а G — адрес многоадресатного IP-получателя или группы получателей
  - (*, G) — совместно используемая древовидная структура, в которой многоадресатная *точка рандеву (Rendezvous Point — RP)* принимает от отправителя многоадресатный трафик и ретранслирует данные получателям, читается как "звездочка запятая G". Звездочка (*) обозначает точку встречи, поскольку она является интерфейсной маской отправителя, который принимает входящие данные от любого реального многоадресатного отправителя. Буква G означает адрес многоадресатного IP-получателя или группы
- Многоадресатные IP-адреса, или адреса класса D, начинаются с цифр 1110 в старших битах адреса. Эти адреса находятся в диапазоне от 224.0.0.0 до 239.255.255.255.
- Узлы, расположенные в любом участке сети, могут зарегистрироваться для подключения к многоадресатной группе, определенной указанным многоадресатным IP-адресом. Регистрация осуществляется посредством протокола IGMP.
- Многоадресатные IP-адреса 224.0.0.1 (все узлы в подсети) и 224.0.0.2 (все маршрутизаторы в подсети) являются фиксированными и не требуют регистрации. Другие фиксированные многоадресатные адреса приведены в приложении Б, "Б.4. стандартные IP-адреса многоадресатного вещания".
- В многоадресатной передаче также используются Ethernet- или MAC-адреса, начинающиеся с 01-00-5e. (Младший значимый бит байта верхнего уровня всегда равен единице.) Многоадресатные IP-адреса должны быть определенным образом преобразованы в многоадресатные MAC-адреса согласно структуре, приведенной на рис. 9.1.
  - 25 старших битов MAC-адресов всегда равны 01-00-5e.
  - 23 младших бита копируются из 23 младших битов IP-адреса.

- Преобразование адресов неуникально. 5 битов IP-адреса не используются. Поэтому 32 различных IP-адреса могут соответствовать одному многоадресатному MAC-адресу.

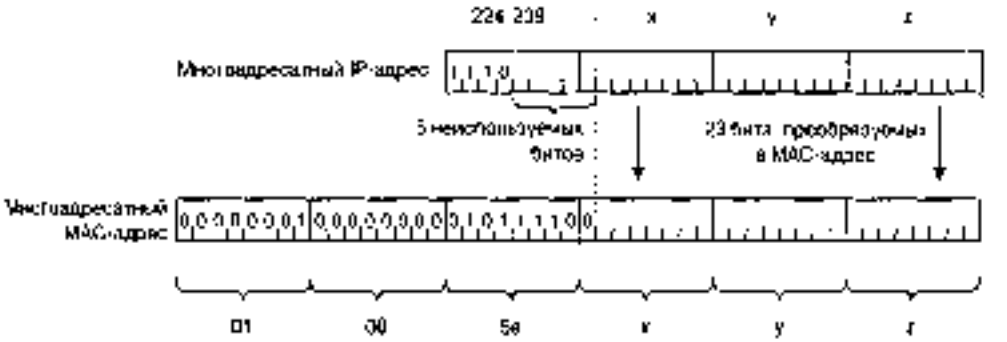


Рис. 9.1. Преобразование многоадресатных адресов

9.1: IGMP-прослушивание

- Некоторые коммутаторы Catalyst могут быть настроены на перехват IGMP-запросов на подключение, которые отправляют узлы для подключения к группам многоадресатного IP-вещания.
- IGMP-запросы на подключение могут возникать в двух ситуациях:
  - узлы отправляют незапрашиваемые отчеты о составе для подключения к определенным многоадресатным группам;
  - многоадресатные маршрутизаторы, функционирующие как *опрашивающие IGMP-узлы (IGMP queriers)*, каждые 60 секунд отправляют IGMP-запросы о составе многоадресатной группы 224.0.0.1, включающей в себя все узлы, в ответ на которые заинтересованные узлы отправляют отчеты о составе для подключения к определенной многоадресатной группе.
- Коммутатор хранит запись о группе многоадресатного IP-вещания, ее MAC-адресе второго уровня и портах коммутатора, к которым подключены запрашивающий узел и многоадресатный маршрутизатор.
- Многоадресатные маршрутизаторы неспособны хранить подробный список всех узлов, принадлежащих какой-либо многоадресатной группе. Вместо этого маршрутизаторы содержат сведения о том, какие многоадресатные группы являются активными в определенных подсетях.
- Коммутатор также ретранслирует первоначальный запрос на подключение для многоадресатной группы всем известным ему многоадресатным маршрутизаторам.
- Если в сети не существует многоадресатные маршрутизаторы, коммутатор может быть настроен для функционирования в качестве опрашивающего IGMP-узла.
- Когда какой-либо узел намерен покинуть многоадресатную группу, протокол IGMP версии 1 обнаруживает только отсутствие отчетов о составе от этого узла. В то же время вторая версия IGMP позволяет узлу в любой момент отправить многоадресатной группе 224.0.0.2, включающей в себя все маршрутизаторы, IGMP-сообщение о выходе из состава своей группы (IGMP leave group message).

- Если коммутатор перехватывает на каком-либо порту IGMP-сообщение о выходе из состава группы, то обычно через тот же порт он отправляет запрос этой многоадресатной группе. Если на отправленный запрос не поступают ответы от узлов и на порту не обнаружены многоадресатные маршрутизаторы, порт удаляется из многоадресатной группы. Чтобы коммутатор удалил порт из многоадресатной группы немедленно после получения сообщения о выходе, следует использовать *IGMP-функцию ускоренной обработки отключений (IGMP Fast-Leave Processing)*.
- Изменения топологии распределенного связующего дерева, происходящие в сети VLAN, также могут приводить к тому, что коммутатор удалит любую информацию о многоадресатном группе, полученную перехватом IGMP-прослушивания. Эта информация должна быть получена повторно.

Совет

IGMP-прослушивание поддерживается на всех коммутирующих платформах Catalyst кроме коммутаторов 2900XL и 3500XL. IGMP-прослушивание для перехвата и изучения каждого многоадресатного IGMP-пакета требует специализированного аппаратного обеспечения, которое недоступно в моделях коммутаторов начального уровня.

Конфигурирование функции

1. Включение IGMP-прослушивания (*необязательно*).

Система COS	`set igmp {enable | disable}`
Система IOS	`ip igmp snooping` (режим конфигурирования интерфейса)

IGMP-прослушивание стандартно включено на всех поддерживающих эту функцию коммутаторах, кроме устройств семейства Catalyst 5000.

2. Разрешение прослушивания другого источника (*необязательно; только для коммутаторов Catalyst 6000/6500*).

Система COS	Нет
Система IOS	`ip igmp snooping mrouter learn {cgmp | pim-dvmrp}` (режим конфигурирования интерфейса)

В дополнение к обычному IGMP-прослушиванию коммутатор также может получать информацию, прослушивая CGMP-сообщения (`cgmp`) или сообщения PIM-DVMRP (`pim-dvmrp`).

3. Использование функции ускоренной обработки отключений (*необязательно*).

Система COS	`set igmp fastleave {enable | disable}`
Система IOS	`ip igmp snooping fast-leave` (режим конфигурирования интерфейса)

Стандартно эта функция отключена. Ускоренное отключение (Fast-Leave) уничтожает параметры запаздывания при удалении многоадресатной группы, однако эту функцию следует использовать только в тех VLAN-сетях, где к каждому порту коммутатора подключен единственный узел.

4. Статическое определение порта многоадресатного маршрутизатора (*необязательно*).

Система COS	`set multicast router` *mod/port*
Система IOS	`ip igmp snooping mrouter {interface {`*interface interface-number*`} | {Port-channel` *number*`}}` (режим конфигурирования интерфейса)

IGMP-прослушивание автоматически обнаруживает порты, к которым подключены многоадресатные маршрутизаторы. Также можно задать статическое определение порта многоадресатного маршрутизатора.

5. Определение статической записи многоадресатного узла (*необязательно*).

Система COS	`set cam {static | permanent}` *{mac address} {mod/port}*
Система IOS	`ip igmp snooping static` *{mac-address} {*`interface` *{*`interface` *interface-number}} | {*`Port-channel` *number}}* (режим конфигурирования интерфейса)

Узел, подключенный к определенному интерфейсу, статически включается в состав многоадресатной группы *mac-address* (в трехблочном формате с точками-разделителями) на текущем VLAN интерфейсе. В COS-коммутаторах статическая запись может использоваться до следующей перезагрузки коммутатора (`static`) или даже после нее (`permanent`).

6. Настройка для работы в качестве опрашивающего узла (*необязательно*)

а) Включение режима опрашивающего узла.

Система COS	`set igmp querier {enable | disable} vlan`
Система IOS	`ip igmp snooping querier` (режим конфигурирования интерфейса)

Стандартно функция опрашивающего IGMP-узла отключена. Она может выполняться коммутатором тогда, когда не существуют другие доступные многоадресатные маршрутизаторы и нет необходимости маршрутизировать многоадресатные пакеты в данной локальной сети. Необходимо использовать ключевое слово `enable` и указать номер VLAN-сети (`vlan`), в которой будет применяться функция.

б) Регулировка интервала запросов (*необязательно*).

Система COS	`set igmp querier vlan qi` *seconds*
Система IOS	`ip igmp query-interval` *seconds* (режим глобальной конфигурации)

Время между общими IGMP запросами, или интервал запросов в сети `vlan`, можно установить равным значению *seconds* (стандартно 125 секунд).

в) Настройка интервала самовыбора (*необязательно*).

Система COS	`set igmp querier vlan qqi` *seconds*
Система IOS	`ip igmp query-timeout` *seconds* (режим глобальной конфигурации)

Если в сети VLAN есть несколько опрашивающих узлов, то будет выбран и останется опрашивающим только один из них. Если в сети VLAN в течение интервала, равного параметру *seconds* (стандартно 300 секунд), не обнаружены другие общие IGMP-запросы, то коммутатор выбирает себя в качестве опрашивающего узла.

Совет

Выборы опрашивающего узла осуществляются на основании IP-адреса отправителя в общих запросах. В определенной VLAN-сети коммутаторы в качестве адреса IGMP-отправителя используют IP-адрес VLAN-интерфейса. Выборы выигрывает узел с наименьшим IP-адресом.

Пример конфигурирования функции IGMP-прослушивания

IGMP-прослушивание глобально включается на COS-коммутаторе и на определенных интерфейсах IOS-коммутатора. Разрешено использование функции ускоренной обработки отключений. Статическая запись для группы многоадресатного IP-вещания 224.100.1.35 (MAC-адрес 01-00-5e-64-01-23) конфигурируется с перечислением портов коммутатора 2/1 и 2/3 в качестве постоянных членов. Эти порты назначены сети VLAN 199.

Система COS
```
set igmp enable
set igmp fastleave enable
set vlan 199 2/1-48
set cam permanent 01-00-5e-64-01-23 2/1,2/3
```

Система IOS
```
interface fastethernet 2/1
(режим глобальной конфигурации)
ip igmp snooping
ip igmp snooping fast-leave
switchport access vlan 199
(три приведенные выше команды вводятся в режиме конфигурирования интерфейса)
interface fastethernet 2/3
(режим глобальной конфигурации)
ip igmp snooping
ip igmp snooping fast-leave
switchport access vlan 199
(три приведенные выше команды вводятся в режиме конфигурирования интерфейса)
interface vlan 199
(режим глобальной конфигурации)
ip igmp snooping static 0100.5364.0123 interface
fastethernet 2/1
ip igmp snooping static 0100.5364.0123 interface
fastethernet 2/3
(две приведенные выше команды вводятся в режиме конфигурирования интерфейса)
```

Отображение сведений о функции IGMP-прослушивания

В табл. 9.2 приведены некоторые команды коммутаторов, которые можно использовать для отображения полезной информации об IGMP-прослушивании.

Таблица 9.2. Команды для отображения информации об IGMP-прослушивании

Функция	Операционная система коммутатора	Команда
IGMP-статистика	COS	`show igmp statistics [vlan-id]`
	IOS	`show ip igmp interface interface interface-number`
Обнаруженные многоадресатные маршрутизаторы	COS	`show multicast router igmp [mod/port] [vlan-id]`
	IOS	`show ip igmp snooping mrouter interface vlan vlan-id`
Количество многоадресатных групп внутри VLAN-сети	COS	`show multicast group count [vlan-id]`
	IOS	`show mac-address-table multicast vlan-id count`
Сведения о многоадресатной группе	COS	`show multicast group [mac-addr ] [vlan-id]` или `show mac multicast {{Vlan vlan-id } grp-mac-addr}]`
	IOS	`show mac-address-table multicast {mac-group-address [vlan-id]}`
IGMP-прослушивание на интерфейсе	COS	Нет
	IOS	`show ip igmp interface vlan-id`

9.2: протокол CGMP

- CGMP — протокол, согласованно действующий между маршрутизаторами и коммутаторами производства корпорации Cisco, который используется для сдерживания многоадресатного трафика.
- Протокол CGMP на коммутаторе Catalyst доверяет CGMP-маршрутизатору предоставление многоадресатных запросов на подключение или отключение, что позволяет осуществлять эффективную регистрацию многоадресатных групп без изучения IGMP-сообщений коммутатором.

- В протоколе CGMP для всех сообщений используется многоадресатный MAC-адрес получателя 01-00-0c-dd-da-dd. Сообщения протокола распространяются лавинно всем портам коммутатора, поэтому даже коммутаторы, не поддерживающие CGMP, ретранслируют его информацию.
- Для работы протокола CGMP требуется отключить IGMP-прослушивание. Эти две функции являются взаимоисключающими. Однако на маршрутизаторе можно настроить оба протокола — как IGMP, так и CGMP — для поддержки различных коммутирующих платформ.
- Протокол CGMP поддерживается на всех коммутирующих платформах Catalyst, кроме коммутатора Catalyst 6000.

Конфигурирование функции

1. Включение протокола CGMP на маршрутизаторе *(только для маршрутизатора)*.

Система IOS	`ip cgmp` (режим конфигурирования интерфейса)

Протокол CGMP необходимо включить на маршрутизаторе, который также осуществляет многоадресатную маршрутизацию. После включения протокола маршрутизатор может отправлять CGMP-сообщения коммутаторам Catalyst.

2. Включение протокола CGMP на коммутаторе.

Система COS	`set cgmp {enable | disable}`
Система IOS	`[no] cgmp` (режим глобальной конфигурации)

Стандартно протокол CGMP отключен на коммутаторах с операционной системой COS, но включен на IOS-коммутаторах. Для включения протокола на COS-коммутаторах необходимо использовать приведенную выше команду.

3. Статическое определение CGMP-маршрутизатора *(необязательно)*.

Система COS	`set multicast router mod/port`
Система IOS	Нет

Стандартно маршрутизаторы, использующие описываемый протокол, объявляют о себе CGMP-коммутаторам. Если необходимо, можно задать статическое определение порта многоадресатного маршрутизатора.

4. Использование CGMP-функции быстрого отключения *(необязательно)*.

Система COS	`set cgmp leave {enable | disable}`
Система IOS	`cgmp leave-processing` (режим глобальной конфигурации)

Функция быстрого отключения от группы (Fast-Leave) стандартно выключена. Если она включена, коммутатор прослушивает IGMPv2-сообщения о выходе из состава группы. Когда сообщение о выходе перехватывается на каком-либо порту и последующие сообщения о подключении отсутствуют, порт отсекается от

многоадресатной группы без какого-либо CGMP-вмешательства со стороны маршрутизатора.

Пример конфигурирования протокола CGMP

Прежде всего необходимо настроить маршрутизатор для поддержки протокола CGMP на его интерфейсе VLAN 199 (см. команды, приведенные ниже).

Система IOS	`interface vlan 199`
	(режим глобальной конфигурации)
	`ip cgmp`
	(режим конфигурирования интерфейса)

В коммутаторе, подключенном к сети VLAN 199, включается поддержка протокола CGMP. Кроме того, для эффективной обработки многоадресатных запросов на отключение (leave requests) включается функция ускоренной обработки отключений.

Система COS	`set cgmp enable` `set cgmp leave enable`

Система IOS	`cgmp`
	`cgmp leave-processing`
	(обе команды вводятся в режиме глобальной конфигурации)

Отображение информации о протоколе CGMP

В табл. 9.3 перечислены команды коммутатора, которые можно использовать для отображения полезных сведений о протоколе CGMP.

Таблица 9.3. Команды для отображения CGMP-информации

Функция отображения	Операционная система коммутатора	Команда
CGMP-состояние	COS	`show cgmp leave`
	IOS	`show cgmp state`
CGMP-статистика	COS	`show cgmp statistics [vlan_id]`
	IOS	Нет
CGMP-маршрутизаторы	COS	`show multicast router cgmp` *[mod/port] [vlan-id]*
	IOS	`show cgmp router` *{address}*
CGMP-группы	COS	`show multicast group` *[mac-addr] [vlan-id]*
	IOS	`show cgmp {vlan` *vlan-id* `| group` *{address}*`}`

9.3: протокол GMRP

- Протокол GMRP является стандартным протоколом управления многоадресатным вещанием, который определен в спецификации IEEE 802.1p.
- Для координации своей многоадресатной активности узел сети использует оба протокола — как IGMP (на третьем уровне), так и GMRP (на втором уровне). Когда узлу нужно подключиться к многоадресатной группе, он отправляет IGMP-запрос на подключение наряду с GMRP-запросом такого же типа.
- Коммутатор перенаправляет управляющие IGMP-пакеты многоадресатному маршрутизатору. GMRP-трафик используется коммутатором для определения того, какие порты коммутатора необходимо добавить в многоадресатную группу.
- Коммутатор периодически опрашивает узлы с помощью GMRP-сообщения о выходе из состава всех групп (GMRP leave-all message). Узлы, которые намерены продолжать участвовать в многоадресатной группе, должны отослать запросы на подключение. В противном случае узлы могут отправить GMRP-сообщение о выходе или просто не отвечать на сообщение коммутатора.

Конфигурирование функции

1. Включение протокола GMRP.

   а) Включение протокола GMRP во всех VLAN-сетях и портах коммутатора.

Система COS	`set gmrp {enable | disable}`
Система IOS	Нет

   б) Включение или отключение GMRP на определенных портах.

Система COS	`set port gmrp` *mod/ports...*`{enable | disable}`
Система IOS	Нет

   После включения протокола GMRP на коммутаторе он включается во всех VLAN-сетях и портах коммутатора. Протокол можно отключить на тех портах, где он не нужен.

2. Идентификация портов, к которым подключены маршрутизаторы.

Система COS	`set gmrp fwdall enable` *mod/port...*
Система IOS	Нет

   Коммутатору необходимы сведения о том, к каким портам подключены какие-либо многоадресатные маршрутизаторы. Такая информация нужна для перенаправления на заданные порты всего многоадресатного трафика. Трафик протокола GMRP не перенаправляется, поскольку он используется только узлами и коммутаторами для сопровождения многоадресатного трафика.

3. Назначение типа GMRP-регистрации.

   Тип регистрации определяет участие порта коммутатора в регистрации узлов многоадресатной группы. В обычном режиме (стандартные установки) узлы могут динамически регистрироваться и покидать многоадресатные группы на

порту. В фиксированном режиме текущая регистрация замораживается; дальнейшее подключение или отключение не разрешается. В запрещенном режиме вся многоадресатная регистрация на порту освобождается, и дальнейшие подключения не допускаются.

Система COS	`set gmrp registration {normal | fixed | forbidden} mod/port..`
Система IOS	Нет

4. Тонкая настройка GARP/GMRP-таймеров (*необязательно*).

Совет

GARP-таймеры используются для определения времени, когда можно отправлять или принимать управляющие сообщения, такие, как запросы на подключение или отключение. Если требуется настроить эти таймеры, это нужно делать последовательно на всех коммутаторах и узловых устройствах.

а) Настройка таймера подключения (*необязательно*)

Система COS	`set gmrp timer join timer-value`
Система IOS	Нет

Таймер подключения используется для задания темпа передачи управляющих GARP-сообщений (например, запросов на подключение). Такие сообщения могут отправляться только через определенные интервалы времени (параметр `timer-value`, от 1 до 2 147 483 647 миллисекунд, стандартно 200 мс).

б) Настройка таймера отключения (*необязательно*)

Система COS	`set gmrp timer leave timer-value`
Система IOS	Нет

Если узел отправляет запрос на отключение, то порт коммутатора потенциально может быть удален из многоадресатной группы. Прежде чем удалить регистрацию, коммутатор определенное время ожидает получения какого-либо запроса на подключение на данном порту. В таймер отключения устанавливается значение `timer value` (от 1 до 2 147 483 647 миллисекунд, стандартно 600 мс). Таймер отключения должен быть по крайней мере в три раза больше таймера подключения.

в) Регулировка таймера выхода из состава всех групп (*необязательно*).

Система COS	`set gmrp timer leaveall timer-value`
Система IOS	Нет

Если в течение интервала `timer-value` (от 1 до 2 147 483 647 миллисекунд, стандартно 10 000 мс, или 10 секунд) коммутатор не получает ответа от зарегистрированного узла, то узел удаляется из всех многоадресатных групп. Значение этого таймера должно быть больше значения таймера отключения.

Отображение информации о протоколе GMRP

В табл. 9.4 перечислены некоторые команды коммутатора, которые можно использовать для отображения полезных сведений о протоколе GMRP

Таблица 9.4. Команды для отображения GMRP-информации

Функция отображения	Операционная система коммутатора	Команда
GMRP-состояние	COS	`show gmrp configuration`
	IOS	Нет
GMRP-статистика	COS	`show gmrp statistics [vlan]`
	IOS	Нет
GARP-таймеры	COS	`show garp timer`
	IOS	Нет

9.4: протокол RGMP

- RGMP является динамическим протоколом, который управляет многоадресатным трафиком к многоадресатным маршрутизаторам.
- При использовании протокола RGMP, чтобы сокращать многоадресатный трафик к заинтересованным узлам, коммутатор должен использовать IGMP-прослушивание.
- Протокол RGMP может обмениваться данными только с маршрутизаторами, в которых используется этот протокол. Маршрутизаторы периодически отправляют коммутаторам hello-сообщения протокола RGMP.
- Маршрутизаторы, заинтересованные в получении трафика для какой-либо многоадресатной группы, отправляют коммутатору RGMP-запрос на подключение. В противном случае коммутатор не перенаправляет многоадресатный трафик маршрутизатору.
- RGMP поддерживает использование разреженного PIM-режима только на многоадресатных маршрутизаторах

Конфигурирование функции

1. Включение на коммутаторе IGMP-прослушивания.

   Необходимые конфигурационные этапы приведены в разделе "9.1: IGMP-прослушивание"

2. Включение на маршрутизаторе разреженного PIM-режима многоадресатной маршрутизации.

   Этапы конфигурирования маршрутизатора описаны в разделе 7.7 книги *Cisco Field Manual: Router Configuration*.

3. Включение на маршрутизаторе протокола RGMP (*только для маршрутизатора*).

Система IOS	`ip rgmp` (режим конфигурирования интерфейса)

Протокол RGMP должен быть включен на интерфейсах маршрутизатора, которые подключены к RGMP-совместимому коммутатору.

4. Включение протокола RGMP на коммутаторе.

Система COS	`set rgmp {enable | disable}`
Система IOS	Нет

Стандартно протокол RGMP отключен.

Отображение информации о протоколе RGMP

В табл. 9.5 перечислены некоторые команды коммутатора, которые можно использовать для отображения полезных сведений о протоколе RGMP.

Таблица 9.5. Команды для отображения RGMP-информации

Функция отображения	Операционная система коммутатора	Команда
RGMP-статистика	COS	`show rgmp statistics [vlan]`
	IOS	Нет
Многоадресатные группы, запрашиваемые RGMP-маршрутизатором	COS	`show rgmp group [mac-addr] [vlan-id]`
	IOS	Нет
Количество многоадресатных групп, запрашиваемых RGMP-маршрутизатором	COS	`show rgmp group count [vlan-id]`
	IOS	Нет

Дополнительная литература

Рекомендуемые источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Протоколы IGMP и CGMP, многоадресатная маршрутизация

Обзор многоадресатной технологии в протоколе IP (Internet Protocol (IP) Multicast Technology Overview) www.cisco.com/warp/public/cc/pd/iosw/prodlit/ipimt_ov.htm.

Beau Williamson, *Developing IP Multicast Networks*, Vol. 1, Cisco Press.

Кеннеди Кларк, Кевин Гамильтон, *Принципы коммутации в локальных сетях Cisco*, ИД "Вильямс", 2003

Протокол GMRP

Стандарт института IEEE 802.1Q — локальные сети, объединенные с помощью виртуальных мостов (IEEE Standard 802.1Q - Virtual Bridged Local Area Networks): `http://standards.ieee.org/getieee802/802.1.html`.

Протокол RGMP

Замечания по конфигурированию протокола RGMP — протокола управления портами маршрутизатора (Configuration Note. RGMP — Router-port Group Management Protocol): `ftp://ftpeng.cisco.com/ipmulticast/config-notes/rgmp.txt`.

В этой главе...

- **10.1: технология SLB.** В разделе описаны конфигурационные этапы обеспечения балансировки нагрузки, создаваемой потоками данных к одной или нескольким группам серверов.
- **10.2: балансировка нагрузки на брандмауэры.** В разделе рассматриваются этапы конфигурирования, необходимые для балансировки трафика, направленного к одной или нескольким группам брандмауэров.
- **10.3: SLB-тесты.** В этом разделе описаны этапы конфигурации для создания проб, с помощью которых тестируются функции групп серверов и брандмауэров.

Глава 10

Балансирование нагрузки на серверы (SLB)

10.1: технология SLB

- С помощью механизма *балансировки нагрузки на серверы* (Server Load Balancing — SLB) обеспечивается IP-адрес виртуального сервера, представляющего группу реальных физических серверов, к которому могут подключаться клиенты. На рис. 10.1 наглядно демонстрируется основная идея службы балансирования нагрузки на серверы. Клиент получает доступ к логическому "виртуальному" серверу (с IP-адресом v.v.v.v), который существует только в SLB-конфигурации коммутатора Catalyst 6000. Группа физических "реальных" серверов (с IP-адресами x.x.x.x, y.y.y.y и z.z.z.z) конфигурируется как серверная группа (server farm). Потоки данных, следующий между клиентами и виртуальным сервером, незаметно для клиентов распределяются среди нескольких реальных серверов.

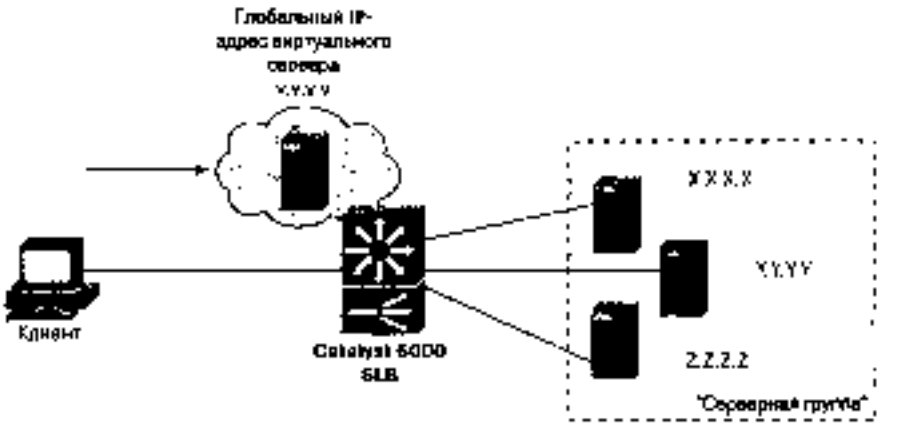


Рис. 10.1. Схема SLB

- Поскольку клиенты открывают новые соединения с виртуальным сервером, в методике SLB решение о том, какой из реальных серверов будет использоваться, принимается на основании алгоритма балансировки нагрузки.
- Балансировка нагрузки на серверы осуществляется одним из описанных ниже методов.
  - **Метод циклического взвешивания** (weighted round-robin). Каждому реальному серверу назначается определенный вес относительно других серверов, который дает ему возможность обрабатывать соединения. Сервер с заданным весом *n* получает *n* новых соединений, прежде чем SLB перешлет нагрузку на следующий сервер.
  - **Метод взвешивания по минимуму соединений** (weighted least connections). Новые соединения назначаются методом SLB определенному реальному серверу с наименьшим числом активных соединений. Каждому реальному серверу назначается вес *m*, причем емкость сервера для активных соединений равна отношению веса *m* к сумме веса всех серверов. Метод SLB назначает новые соединения реальному серверу с наибольшим запасом емкости по активным соединениям.
- Служба SLB с помощью метода взвешивания по минимуму соединений управляет доступом к новому реальному серверу, обеспечивая медленный запуск (slow start function). Новые соединения ограничены по скорости, и допускается постепенное увеличение, что предохраняет сервер от перегрузки.
- Виртуальный сервер может выполнять функцию маскировки (masquerade) IP-адреса для всех TCP- и UDP-портов реальной серверной группы. Также виртуальный сервер может представляться как IP-адрес одного порта или службы серверной группы.
- "Залипание" (sticky) соединения позволяет SLB назначать новые соединения от клиента к тому реальному серверу, который использовал этот клиент последним.
- Методика SLB способна обнаружить сбой реального сервера путем мониторинга нарушенных TCP-соединений. SLB позволяет отключить вышедший из строя сервер от обслуживания и подключить его, как только он снова станет работоспособным.
- В методике SLB для преобразования реальных и виртуальных адресов серверов в том случае, если они расположены в различных подсетях третьего уровня, может применяться *трансляция сетевых адресов серверов (server Network Address Translation — NAT)*.
- В SLB также может использоваться *NAT-трансляция адресов клиентов (client NAT)* для преобразования адресов отправителей клиентских запросов в адреса на серверной стороне SLB-устройства. Эта функция применяется в ситуациях, когда несколько SLB-устройств функционируют так, что возвращаемый трафик может быть отправлен к соответствующему SLB-устройству.
- Методика SLB предоставляет реальным серверам механизм контроля над входящими TCP SYN-лавинами, что позволяет предотвратить определенные типы атак "отказ в обслуживании".
- Методика SLB способна сосуществовать с протоколом HSRP *(Hot Standby Router Protocol — протокол резервного маршрутизатора)* для обеспечения "кумулятивного

резервирования" (stateless backup). Когда один SLB-маршрутизатор выходит из строя, избыточный маршрутизатор может взять на себя SLB-функцию. Однако существующие SLB-соединения будут потеряны и потребуется их повторная установка со стороны клиента.

- Служба SLB операционной системы IOS может также функционировать как диспетчер DFP-балансировки нагрузки (*Dynamic Feedback Protocol — протокол динамической обратной связи*). DFP-диспетчер накапливает сведения о емкости, поступающие от DFP-агентов, запущенных на реальных серверах.

Конфигурирование функции

Внимание!

В этой главе SLB-команды для коммутаторов Catalyst 6000 с собственной системой IOS отмечены аббревиатурой "IOS", а команды для модуля коммутации по содержимому коммутатора Catalyst 6000 — аббревиатурой "CSM". Методика SLB недоступна в COS-платформах, поэтому этот тип формата команд пропущен.

CSM-команды фактически используются для интерфейса командной строки (*Command-Line Interface — CLI*) собственной IOS в режиме конфигурации CSM-модуля. Ниже IOS- и CSM-команды представлены рядом для сравнения. CSM-команды соответствуют программному образу CSM 2.1.

1. Указание виртуальных локальных сетей (VLAN) с клиентской и серверной стороны (только для CSM-модуля)

   а) Начало конфигурирования CSM-модуля

IOS-команда	Нет
CSM-команда	`module csm` *slot-number* (режим глобальной конфигурации)

   В CLI-интерфейсе собственной IOS режим конфигурирования CSM запускается для модуля, расположенного в гнезде с номером *slot-number* шасси коммутатора. Для выхода из этого режима используется команда `exit`. Чтобы определить номер соответствующего гнезда, применяется команда `show module all`.

   б) Указание типа VLAN-сети.

IOS-команда	Нет
CSM-команда	`vlan` *vlan id* `{client	server}`

   Номер VLAN-сети задается параметром *vlan-id* (от 2 до 4095; использование сети VLAN 1 невозможно). Эта сеть уже должна быть определена в базе данных VLAN на коммутаторе. Тип VLAN-сети — `client` или `server` — определяет, где с точки зрения модуля CSM расположены клиенты или серверы (серверная группа реальных серверов). Прежде чем можно будет соответствующим образом использовать CSM-модуль, необходимо определить VLAN-сети на *обеих* сторонах — как на клиентской, так и на серверной. Клиенты и серверы должны быть расположены в различных VLAN-сетях.

в) Назначение основного (primary) IP-адреса (*необязательно*).

IOS-команда	Нет
CSM-команда	`ip address` *ip-address netmask* (режим конфигурирования сети VLAN)

В модуле CSM для каждой VLAN-сети может быть определен один IP адрес. Этот адрес используется для административного трафика (например, для проб) и ARP-запросов.

г) Назначение второстепенного IP-адреса (*необязательно*).

IOS-команда	Нет
CSM-команда	`alias` *ip address netmask* (режим конфигурирования сети VLAN)

Вторичные (secondary) IP-адреса позволяют модулю CSM без использования маршрутизатора обмениваться данными с серверами, расположенными в различных IP-сетях.

д) Выбор стандартного шлюза (*необязательно*).

IOS команда	Нет
CSM-команда	`gateway` *ip-address* (режим конфигурирования сети VLAN)

Стандартный шлюз следующего транзитного перехода или адрес маршрутизатора задается с помощью параметра *ip-address*. Эту команду можно повторять для указания до 7 шлюзов для каждой VLAN-сети или до 255 шлюзов для модуля CSM. Шлюзы обычно используются во VLAN-сети клиентской стороны, хотя, если требуется, они также могут использоваться и на серверной стороне.

е) Указание статических маршрутов для достижения дальних сетей (*необязательно*).

IOS-команда	Нет
CSM-команда	`route` *ip-address netmask* `gateway` *gw-ip-address* (режим конфигурирования сети VLAN)

Статический маршрут может определяться, когда CSM-модулю необходимо получить сведения о том, как достичь серверов, расположенных на расстоянии более одного транзитного перехода. Маршрут определяется параметрами *ip-address* и *netmask* с использованием адреса шлюза *gw-ip-address*. Шлюз должен быть размещен в той же локальной сети, что и CSM-сеть VLAN.

ж) Повторение этапов с 1.б по 1.е для каждой VLAN-сети клиентской и серверной сторон.

з) Определение отказоустойчивой VLAN-сети для резервных CSM-модулей (*необязательно*).

- Идентификация отказоустойчивой VLAN-сети

IOS-команда	Нет
CSM-команда	`vlan vlan-number ft` (режим конфигурирования сети VLAN)

Сеть VLAN следует определять для обоих резервных модулей CSM. Она должна быть частной сетью VLAN, соединяющей два модуля так, чтобы они могли совместно использовать соединения и трафик через резервные каналы. Для каждой пары резервных CSM-модулей должен использоваться отдельный номер частной отказоустойчивой сети VLAN.

- Указание отказоустойчивой группы.

IOS-команда	Нет
CSM-команда	`ft group` *group-id* `vlan` *vlan-id* (режим конфигурирования сети VLAN)

Каждому из избыточных CSM-модулей необходимо задать общий идентификатор отказоустойчивой группы (*group-id*, от 1 до 254). Отказоустойчивой VLAN является сеть с номером *vlan-id* (от 2 до 4095).

- Установка приоритета CSM-модуля (*необязательно*).

IOS-команда	Нет
CSM-команда	`priority` *value* (режим конфигурирования сети VLAN)

Главным CSM становится модуль с наивысшим значением приоритета (параметр *value* — от 1 до 254; стандартно 10).

- Восстановление модуля CSM в качестве главного (*необязательно*).

IOS-команда	Нет
CSM-команда	`preempt`

Стандартно вышедший из строя главный модуль CSM после восстановления не может снова стать главным. Чтобы перевести восстановленный модуль в режим главного, используется команда `preempt`, которую нужно ввести на обоих резервных CSM-модулях, допускающих приоритетное прерывание обслуживания.

- Установка интервала пульсаций (heartbeat interval) (*необязательно*).

IOS-команда	Нет
CSM-команда	`heartbeat-time` *heartbeat-time*

Между избыточными CSM-модулями в отказоустойчивой сети происходит обмен сообщениями пульсации через регулярные промежутки времени (параметр *heartbeat-time* — от 1 до 65 535 секунд; стандартно 1 секунда).

- Установка времени восстановления после сбоев (*необязательно*).

IOS-команда	Нет
CSM-команда	`failover` *failover-time*

Резервный CSM-модуль, прежде чем взять на себя функции главного, после получения последнего сообщения пульсации ожидает в течение определенного времени (параметр `failover-time` — от 1 до 65 535 секунд; стандартно — 3 секунды).

2. Определение серверной группы.

   а) Назначение имени серверной группе.

IOS-команда	`ip slb serverfarm` *serverfarm-name* (режим глобальной конфигурации)
CSM-команда	`serverfarm` *serverfarm-name*

   Серверная группа идентифицируется по имени (параметр *serverfarm-name*) — текстовая строка длиной до 15 символов.

   б) Выбор для серверной группы алгоритма балансировки нагрузки (*необязательно*).

IOS-команда	`predictor {roundrobin | leastconns}`

CSM-команда	`predictor {roundrobin | leastconns | hash url`
	`hash address {source | destination} [ip`
	netmask`] | forward}`

   В методике SLB реальный сервер выбирается с помощью алгоритма `roundrobin` (взвешенного циклического алгоритма выбора) или `leastconns` (взвешенного по минимуму соединений).

   Балансировка нагрузки в модуле CSM также может осуществляться на основании параметра `hash url` (хэш-значение, полученное из URL-строки, используется наряду с командой `url-hash`, см. пункт 1 этапа 5), `hash address` (хэш-значение, полученное с адреса отправителя (`source`) или получателя (`destination`). Параметр *ip-netmask* может определять биты адреса, используемые для хэширования, при этом стандартным значением является маска 255.255.255.255 или "все биты"), или в режиме `forward` (пересылка трафика согласно таблицам маршрутизации модуля CSM).

   в) Использование NAT-преобразования адреса сервера (*необязательно*)

IOS-команда	`nat server` (режим конфигурирования блока серверов)
CSM-команда	`nat server` (режим конфигурирования блока серверов)

   Стандартные адреса виртуального и реального серверов должны быть смежными на втором уровне. Иными словами, служба SLB пересылает пакеты между виртуальным и реальным серверами, заменяя корректные MAC-адреса. Вместо этого можно применять серверное NAT-преобразование, которое допускает использование виртуальным и реальным серверами адресов из отдельных IP-подсетей. Затем служба SLB для пересылки пакетов между виртуальным и реальным серверами заменяет адреса третьего уровня, т.е. IP-адреса, что позволяет разделять серверы посредством нескольких маршрутных переходов.

   г) Использование NAT-преобразования адресов клиентов (*необязательно*).

- Определение пула NAT-адресов.

IOS-команда	`ip slb natpool` *pool-name start-ip end-ip* {`netmask` *netmask* | `prefix-length` *leading-1-bits*} [`entries` *init-addr* [*max-addr*]] (режим глобальной конфигурации)
CSM-команда	`natpool` *pool-name start-ip end-ip* {`netmask` *netmask* | `prefix-length` *leading-1-bits*} (режим глобальной конфигурации)

Пул IP-адресов задается параметром *pool-name* (строка длиной до 15 символов), содержащим диапазон адресов, ограниченный начальным и конечным адресами (параметры *start-ip*, *end-ip*). Связанная с диапазоном маска подсети может быть задана как обычная маска подсети *netmask* (в формате *x.x.x.x*) или как количество старших, равных единице, битов в маске (параметр *leading-1-bits*, от 1 до 32).

Для службы SLB операционной системы IOS с помощью клиентской NAT-трансляции выделяется некоторое количество записей в виде IP-адресов и номеров портов, значение *init-addr* (от 1 до 1 000 000, стандартно — 8000) применяется как первоначально используемый набор. Когда количество динамически выделенных записей достигает половины первоначального числа, выделяются дополнительные записи. Максимальное количество NAT-записей может быть определено параметром *max-addr* (от 1 до 8 000 000, стандартно — количество доступных портов, умноженное на размер диапазона, либо значение в диапазоне 11 000–65 000, или 54 535). Номера портов для NAT-преобразования начинаются с 11 000.

- Активизация пула для NAT-трансляции адресов клиентов.

IOS-команда	`nat client` *pool-name* (режим конфигурирования блока серверов)
CSM-команда	`nat client` *pool-name* (режим конфигурирования блока серверов)

NAT-пул в службе SLB определяется параметром *pool-name* (до пятнадцати символов длиной)

д) Назначение уникального идентификатора для протокола DFP (*необязательно*).

IOS-команда	`bindid` [*bind-id*] (режим конфигурирования блока серверов)
CSM-команда	`bindid` [*bind-id*] (режим конфигурирования блока серверов)

Иногда реальный сервер закрепляется за несколькими серверными группами. Параметр *bind-id* (от 0 до 65533, стандартно — 0) является произвольным идентификационным значением, заданным серверной группе. Каждый экземпляр ссылки реального сервера на это значение позволяет протоколу DFP назначать ей уникальный вес.

е) Проверка сервера с помощью запросов (*необязательно*).

IOS-команда	`probe name` (режим конфигурирования блока серверов)
CSM-команда	`probe` *name* (режим конфигурирования блока серверов)

Запрос, определяемый именем (параметр *name* — текстовая строка длиной до пятнадцати символов), периодически тестирует связь и функционирование сервера. В операционной системе IOS службе SLB предоставлены различные виды запросов: ping, **HTTP** и WSP (*Wireless Session Protocol — протокол беспроводной связи*). В модуле CSM также доступны TCP-, FTP-, Telnet- и DNS-запросы. Более подробная информация по конфигурированию проб приведена в разделе "10.3. SLB-тесты".

ж) Сброс соединения к вышедшему из строя серверу (*необязательно; только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`failaction purge` (режим конфигурирования блока серверов)

Стандартно соединения не сбрасываются при отказе сервера (по `failaction purge`). При использовании модуля CSM для балансирования нагрузки VPN-соединений эту команду необходимо использовать для того, чтобы существующие туннельные соединения к отказавшему серверу отключались автоматически. В противном случае соединения будут находиться в режиме ожидания до тех пор, пока удаленная сторона не получит уведомление об отказе сервера.

з) Указание виртуального сервера перенаправления, принимающего перенаправленный трафик (*необязательно; только для модуля CSM*)

- Имя виртуального сервера перенаправления.

IOS-команда	Нет
CSM-команда	`redirect-vserver` *name* (режим конфигурирования блока серверов)

Виртуальный сервер перенаправления получает имя, заданное в параметре *name* (текстовая строка длиной до пятнадцати символов).

- Назначение адреса и порта для виртуального сервера.

IOS-команда	Нет
CSM-команда	`virtual` *ip address* `tcp` *port* (режим конфигурирования сервера перенаправления)

Виртуальный сервер перенаправления принимает к IP-адресу (*ip address*, стандартно — 0.0.0.0 или пересылка пакетов не используется) и номеру TCP-порта (*port*).

- Запрет клиентского доступа (*необязательно*)

IOS-команда	Нет
CSM-команда	`client` *ip-address [network-mask]* `[exclude]` (режим конфигурирования сервера перенаправления)

Клиентам, которые имеют IP-адреса, попадающие в заданный параметрами *ip-address* (стандартно — 0.0.0.0 или все адреса) и *network-mask* (стандартно — 255.255.255.255 или все сети) диапазон, будет разрешено подключаться к данному виртуальному серверу. Маска *network mask* в этом случае сходна с маской списка доступа, в котором нулевой бит игнорируется, а нулевой — совпадает. Вместо этого в модуле CSM можно использовать ключевое слово **exclude**, позволяющее исключить клиентов с совпадающими IP-адресами.

- Запрет доступа к VLAN-сети (*необязательно*).

IOS-команда	Нет
CSM-команда	`vlan {vlan-number	all}` (режим конфигурирования сервера перенаправления)

При необходимости доступ к виртуальному серверу получат только узлы из указанной сети VLAN (номер *vlan-number* от 2 до 4095) или всех VLAN-сетей (**all** — стандартная установка).

- Объявление о виртуальном сервере (*необязательно*)

IOS-команда	Нет
CSM-команда	**advertise [active]** (режим конфигурирования сервера перенаправления)

Стандартно служба SLB создает статический маршрут (маска сети 255.255.255.255) для адреса данного виртуального сервера к несуществующему логическому интерфейсу Null0. Этот статический маршрут может перераспределяться и анонсироваться каким-либо протоколом маршрутизации. Использование ключевого слова **active** приводит к тому, что маршрут будет анонсироваться только в том случае, если существует по крайней мере один реальный сервер. Объявление можно отключить с помощью ключевых слов **no advertise**, препятствующих созданию статического маршрута.

- Включение резервирования соединений между несколькими модулями CSM (*необязательно*).

IOS-команда	Нет
CSM-команда	**replicate csrp** (режимы конфигурирования сервера перенаправления)

Сведения по соединениям дублируются на остальные CSM-модули, сконфигурированные для поддержки избыточности.

- Удержание соединения после прекращения активности (*необязательно*).

IOS-команда	Нет
CSM-команда	**idle** *duration* (режим конфигурирования сервера перенаправления)

Когда служба SLB обнаруживает отсутствие пакетов для какого-либо соединения, это соединение в течение определенного времени (параметр *duration* по

жит в диапазоне 4-65535 секунд, стандартно 3600 секунд, или 1 час) остается активным, прежде чем будет отправлено TCP-сообщение *Reset* (RST).

- Перенаправление и пересылка сообщений SSL (*необязательно*).

IOS-команда	Нет
CSM-команда	`ssl {https | ftp | ssl-port-number}` (режим конфигурирования сервера перенаправления)

Стандартно HTTP-запросы не пересылаются SSL-серверам. Их можно перенаправлять с помощью ключевых слов `https` (на порт 443), `ftp` или указывая явным образом номер порта (от 1 до 65535) в параметре *ssl-port-number*.

- Отправка строки перемещения (relocation string) для перенаправленных HTTP-запросов (*необязательно*).

IOS-команда	Нет
CSM-команда	`webhost relocation relocation-string [301 | 302]` (режим конфигурирования сервера перенаправления)

Строка перемещения (*relocation-string* — текстовая строка длиной до 127 символов) отправляется в ответ на перенаправленный HTTP-запрос. К исходному URL-пути может быть добавлена строка перемещения путем редактирования строки с переменной `%p`. Возвращаемый код состояния может принимать два значения — `301` ("запрашиваемый ресурс получил новый постоянный URL адрес") или `302` (стандартный код: "запрашиваемый ресурс временно расположен по другому URL-адресу").

- Отправка строки перемещения, когда сервер перенаправления отключен (*необязательно*).

IOS-команда	Нет
CSM-команда	`webhost backup backup-string [301 | 302]` (режим конфигурирования сервера перенаправления)

Строка *backup-string* отправляется в ответ на перенаправленный HTTP-запрос в случае, если нет доступных реальных серверов. К исходному URL-пути может быть добавлена строка перемещения путем редактирования строки с переменной `%p`. Возвращаемый код состояния может принимать два значения — `301` ("запрашиваемый ресурс получил новый постоянный URL-адрес") или `302` (стандартный код: "запрашиваемый ресурс временно расположен по другому URL-адресу").

- Включение виртуального сервера перенаправления.

IOS-команда	Нет
CSM-команда	`inservice` (режим конфигурирования сервера перенаправления)

3. Указание одного или нескольких реальных серверов в серверной группе

а) Идентификация реального сервера.

IOS-команда	`real` *ip-address* (режим конфигурирования блока серверов)
CSM-команда	`real` *ip-address* [*port*] (режим конфигурирования блока серверов)

Реальный сервер имеет IP-адрес, заданный параметром *ip-address*. Модуль CSM позволяет преобразовать номера портов для сервера путем указания номера порта (*port*, от 1 до 65535, стандартно преобразование портов отключено).

б) Указание порогового количества соединений (*необязательно*).

- Установка максимального количества соединений

IOS-команда	`maxconns` *number* (режим конфигурирования свойств сервера)
CSM-команда	`maxconns` *number* (режим конфигурирования свойств сервера)

В любое заданное время можно ограничить количество активных соединений реального сервера (*number*, от 1 до 4 294 967 295 соединений; стандартно — 4 294 967 295).

- Установка минимального порогового количества соединений (*необязательно только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`minconns` *number* (режим конфигурирования свойств сервера)

При применении этого порога ключевое слово `minconns` с параметром *number* (от 1 до 4 294 967 295 соединений) устанавливает пороговое значение, т.е. количество активных соединений, которые должны быть разорваны, прежде чем снова будет разрешено создавать новые соединения.

в) Назначение весового коэффициента относительной емкости (*необязательно*)

IOS-команда	`weight` *weighting-value* (режим конфигурирования свойств сервера)
CSM-команда	`weight` *weighting-value* (режим конфигурирования свойств сервера)

Реальный сервер получает значение веса (*weighting-value*, от 1 до 255 для IOS, от 1 до 100 для CSM, стандартно — 8), которое указывает его емкость относительно других реальных серверов в серверной группе. Для взвешенного циклического метода значение *weighting-value* определяет количество последовательных соединений, которое получит сервер до того, как SLB перейдет к следующему серверу. Для взвешенного метода по минимуму соединений следующее соединение передается серверу, количество соединений которого наименее близко к емкости по сравнению с другими серверами. Емкость ви-

числяется как значение веса, разделенное на сумму весов всех реальных серверов в данной серверной группе.

г) Переназначение соединений при отсутствии ответа сервера (*необязательно; только в IOS SLB*).

IOS-команда	`reassign` *threshold* (режим конфигурирования свойств сервера)
CSM-команда	Нет

Служба SLB пытается назначить новое соединение к реальному серверу путем перезаписи первоначального клиентского SYN-пакета. Если сервер не ответит SYN-квитированием (handshake) до того, как клиент повторно отправит свой SYN-запрос, SYN-пакет, на который не поступил ответ, записывается. По достижении порогового количества (*threshold*, от 1 до 4; стандартно — 3) остающихся без ответа SYN-запросов функция SLB переназначает соединение следующему серверу.

д) Определение порога аварийного сервера (*необязательно; только в IOS SLB*).

IOS-команда	`faildetect numconns` *number-conns* [`numclients` *number clients*] (режим конфигурирования свойств сервера)
CSM-команда	Нет

Сервер считается аварийным, если другому серверу переназначается определенное количество его TCP-соединений (параметр *number-conns*, от 1 до 255; стандартно — 8 соединений). Также можно использовать ключевое слово `numclients` для указания количества уникальных клиентов (*number-clients*, от 1 до 8, стандартно — 2), соединения которых были разорваны.

е) Указание периода времени до повторного запроса аварийного сервера (*необязательно, только в IOS SLB*).

IOS-команда	`retry` *retry-value* (режим конфигурирования свойств сервера)
CSM-команда	Нет

По истечении времени, указанного в параметре *retry value* (от 1 до 3600 секунд; стандартно 60 секунд), после того как реальный сервер объявлен "аварийным", функция SLB пытается назначить новое подключение к нему. Чтобы избежать этого, можно использовать значение 0.

ж) Прием перенаправленного HTTP-трафика (*необязательно, только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`redirect-vserver` *name* (режим конфигурирования свойств сервера)

Реальный сервер может получить трафик, перенаправленный виртуальному серверу перенаправления, имя которого указывается в команде (*name* — тексто-

вая строка длиной до пятнадцати символов). Виртуальный сервер конфигурируется на этапе 2.

з) Разрешение на использование функцией SLB реального сервера

IOS-команда	`inservice` (режим конфигурирования свойств сервера)
CSM команда	`inservice` (режим конфигурирования свойств сервера)

Стандартно, если реальный сервер отключен, он не используется функцией SLB. Чтобы отключить сервер, используется команда `no inservice`.

4. Определение стратегии балансировки нагрузки для специфического трафика (*только для модуля CSM*).

а) Указание таблицы преобразования URL-адресов (*необязательно*)

- Указание имени таблицы.

IOS-команда	**Нет**
CSM-команда	**map** `url-map-name url`

Таблица преобразования с именем `url-map-name` (текстовая строка длиной до пятнадцати символов) содержит условия отбора URL-адресов.

- Отбор URL-адресов на основании регулярного выражения (regular expression).

IOS-команда	Нет
CSM-команда	`match protocol http url urln`

Регулярное выражение `urln` (до 255 символов) сравнивается с содержимым адреса URL. В одну таблицу преобразования можно ввести до 1023 **match**-директив.

Регулярное выражение может содержать ряд символов для отбора адресов URL: * (ни одного или несколько символов), ? (один символ), \ (escape-символ), [ ] (диапазон символов, заданный с помощью type), ^ (не совпадает с любым символом диапазона, указанного после символа), .\a (сигнал (alert), или символ ASCII 7), .\b (backspace, или ASCII 8), .\f (подача (form-feed), или ASCII 12), .\n (новая строка, или ASCII 10), .\r (возврат каретки, или ASCII 13), .\t (табулятор, или ASCII 9), .\v (пуль, или ASCII 0), .\\ (обратная косая черта) и .\### (любой символ, задаваемый двумя шестнадцатеричными числами для ASCII-значения).

б) Задание таблицы для проверки cookie-файлов[1]

- Указание имени таблицы

IOS-команда	Нет
CSM-команда	**map** `cookie-map name cookie`

[1] *В сети Internet — небольшой фрагмент данных о предыстории обращений пользователя к конкретному WWW-серверу, автоматически создаваемый браузером по запросу пользователя — Прим. ред.*

В таблице с именем *cookie-map-name* (текстовая строка длиной до пятнадцати символов) содержится одно или несколько условий проверки cookie-файлов.

- Проверка cookie-файлов.

IOS-команда	Нет
CSM-команда	`match protocol http cookie` *cookie-name* `cookie-value` *cookie-value-expression*

Cookie-файлы проверяются по имени *cookie-name* (текстовая строка длиной до 63 символов) и регулярному выражению *cookie value expression* (текстовая строка длиной до 255 символов). Символы регулярных выражений определяются на этапе 4.а. В таблице проверки cookie-файлов может использоваться множество команд `match`, но все они должны быть выполнены до того, как можно будет проверить cookie.

н) Определение таблицы для проверки строк заголовков.

- Указание названия таблицы.

IOS-команда	Нет
CSM-команда	`map` *header-map-name* `header`

Таблица с именем *header-map-name* (строка длиной до пятнадцати символов) содержит одно или несколько условий проверки **HTTP**-заголовков.

- Совпадение HTTP-заголовков.

IOS-команда	Нет
CSM команда	`match protocol http header` *field* `header-value` *expression*

HTTP-заголовки проверяются по литеральному полю с именем *field* (текстовая строка длиной до 63 символов) и регулярному выражению (*expression* — текстовая строка длиной до 127 символов). Символы регулярных выражений определяются на этапе 4.а. В таблице проверки заголовков может использоваться множество команд `match`, но все они должны быть выполнены до того, как можно будет проверить заголовок.

г) Проверка установленных соединений от клиента (*необязательно*).

IOS-команда	Нет
CSM команда	`sticky` *sticky group id* {`netmask` *netmask* | `cookie` *name* | `ssl`} [`timeout`]

Соединения от клиента, которые соответствуют правилу, могут быть переведены в режим "sticky" (режим привязки к определенному объекту, т.н. "залипание"), так как все они используют один и тот же реальный сервер. Общие команды `sticky` следует группировать в `sticky`-группу. *sticky-group-id* (от 1 до 255). Можно закрепить `sticky`-соединения на основании замаскированного IP-адреса клиента (`netmask` *netmask* — стандартная маска подсети), имени `cookie`-файла (`cookie`

name) или данных протокола SSL (**ssl**). Режим "залипших" соединений позволяет на определенное время (**sticky** *time* — от 0 до 65535 минут; стандартно — 1440 минут или 24 часа; 0 отключает **sticky**-режим) запоминать последний использованный реальный сервер.

з) Проверка адреса отправителя клиента (*необязательно*).

- Создание именованного стандартного списка IP-доступа.

IOS-команда	Нет
CSM-команда	**ip access-list standard** *access-list-name*

В этой команде задается список доступа с именем *access-list-name* (текстовая строка).

- Разрешение или запрет адресов отправителей.

IOS-команда	Нет
CSM-команда	{**permit** | **deny**} *source-address* [*source-wildcard*] (режим конфигурирования списка доступа)

Клиентские адреса отправителей можно разрешать (**permit**) или запрещать (**deny**) на основании параметра *source-address* (в формате IP-адреса) и маски подсети (*source-wildcard* — формат маски похожий, но первые биты используются в качестве инвертированной маски). В список доступа можно включать одну или несколько команд **permit** и **deny**. Они обрабатываются в том порядке, в котором были введены.

е) Создание одной или нескольких стратегий балансировки нагрузки.

- Указание имени правила (policy).

IOS-команда	Нет
CSM-команда	**policy** *policy-name*

Правилу назначается имя *policy-name* (текстовая строка длиной до пятнадцати символов). В модуле CSM можно сконфигурировать до 12 287 различных правил.

Совет

Правило может содержать одну или несколько последовательных команд **map** и **group**. Если в одной из них используется множество карт и групп правил, проверяемый трафик должен соответствовать им всем.

- Использование URL-карты (*необязательно*).

IOS-команда	Нет
CSM-команда	**url-map** *url-map-name* (режим конфигурирования правил)

URL-адреса, проверяемые URL-таблицей с именем *url-map-name*, обрабатываются согласно этому правилу. Таблицы URL создаются на этапе 4,а.

- Использование таблицы cookie-файлов (*необязательно*).

IOS-команда	Нет
CSM-команда	**cookie-map** *cookie-map-name* (режим конфигурирования правил)

Согласно этой стратегии обрабатываются cookie-файлы, проверяемые с помощью таблицы с именем *cookie-map-name*. Таблицы cookie-файлов создаются на этапе 4,б.

- Использование таблицы заголовков (*необязательно*).

IOS-команда	Нет
CSM-команда	**header-map** *header-map name* (режим конфигурирования правил)

Этим правилом контролируются заголовки, соответствующие таблице с именем *header-map-name*. Таблицы заголовков создаются на этапе 4.в.

- Использование sticky группы (группы закрепленных соединений, *необязательно*).

IOS-команда	Нет
CSM-команда	**sticky-group** *group-id* (режим конфигурирования правил)

Соединения, соответствующие номеру sticky-группы (*group-id* — от 1 до 255; стандартно — 0, т.е. "липкие" соединения не используются) в этой группе правил, отправляются одному и тому же реальному серверу. Sticky группы создаются на этапе 4,г.

- Использование группы клиентского фильтра (*необязательно*).

IOS-команда	Нет
CSM-команда	**client-group** {*acl number* | *acl name*} (режим конфигурирования правил)

Этим правилом контролируется трафик, соответствующий стандартному списку IP-доступа с номером *acl-number* или именем *acl-name*. Списки доступа клиентского фильтра создаются на этапе 4,д.

- Маркирование трафика DSCP-значениями (*необязательно*).

IOS команда	Нет
CSM-команда	**set ip dscp** *dscp-value* (режим конфигурирования правил)

Пакеты, соответствующие этому правилу, имеют собственные DSCP значения (*Differentiated Service Code Point — поле кодирования дифференцированных служб*), равные *dscp-value* (от 0 до 63, стандартного значения нет). Более подробная информация по DSCP-значениям приведена в разделе "13.1: теоретические основы механизмов обеспечения качества обслуживания"

- Привязка серверной группы к определенному правилу.

IOS-команда	Нет
CSM-команда	**serverfarm** *serverfarm-name* (режим конфигурирования правил)

Каждый набор правил используется на виртуальных серверах, но в первую очередь он должен быть связан только с одной серверной группой, что позволит ему управлять балансировкой нагрузки на реальные серверы в серверной группе с именем *serverfarm-name* (текстовая строка).

5. Указание виртуального сервера для серверной группы.

а) Указание имени виртуального сервера.

IOS-команда	**ip slb vserver** *virtual-server-name* (режим глобальной конфигурации)
CSM-команда	**vserver** *virtual-server name*

Виртуальному серверу задается имя *virtual-server-name* (текстовая строка длиной до пятнадцати символов).

б) Назначение серверной группе виртуального сервера

IOS-команда	**serverfarm** *serverfarm-name* (режим конфигурирования виртуального сервера)
CSM-команда	**serverfarm** *serverfarm name* (режим конфигурирования виртуального сервера)

В службе SLB виртуальный сервер используется в качестве клиентской части (front end) серверной группы с именем *serverfarm-name* (текстовая строка длиной до пятнадцати символов)

в) Указание возможностей виртуального сервера.

IOS-команда	**virtual** *ip-address [network-mask]* {**tcp** | **udp**} [*port* | **wap** | **wap-wtls** | **wap-wtp-wtls**] [**service** *service-name*] (режим конфигурирования виртуального сервера)
CSM-команда	**virtual** *ip-address [network-mask]* {**tcp** | **udp** | **any** | *protocol-number*} **port** [**service ftp**] (режим конфигурирования виртуального сервера)

Виртуальный сервер представлен IP-адресом (*ip-address*, стандартно — 0.0.0.0 или "все сети") с маской (*network-mask*; стандартно — 255.255.255.255).

В операционной системе IOS службой SLB обеспечивается балансировка нагрузки на определенные порты (**tcp** или **udp** *port*), например: **dns** или **53** (Domain Name System — система имен доменов), **ftp** или **21** (File Transfer Protocol — протокол передачи файлов), **https** или **443** (HTTP по протоколу Secure Socket Layer, SSL), **www** или **80** (протокол HTTP), **telnet** или **23** (служба Telnet), **smtp** или **25** (протокол SMTP), **pop3** или **110** (протокол POP

версии 3), pop2 или 109 (протокол POP версии 2), nntp или 119 (Network News Transport Protocol — протокол передачи сетевых новостей), mobile-ip или 350 (Mapping of Airline Traffic over IP, type A — преобразование авиатранспортной информации на основе IP-протокола, тип A). Порт номер 0 задается для того, чтобы указать, что данный виртуальный сервер принимает соединения на всех портах.

Другими альтернативами номеру порта являются значения wap (не требующий установки соединения протокол WSP, порт 9200), wap-wtp (ориентированный на соединение WSP, порт 9201 со службой WAP FSM), wap-wtls (не требующий установки соединения безопасный протокол WSP, порт 9202) и wap-wtp-wtls (безопасный протокол с установлением соединения WSP, порт 9203).

Для модуля CSM в качестве протокола можно указать tcp, udp или any (любой протокол, номер порта не требуется; стандартная установка) либо число от 0 до 255 (*protocol-number*). Может быть задано любое допустимое имя или номер порта (*port*, от 0 до 65535).

Можно использовать ключевое слово service, которое вынуждает службу SLB назначать все соединения, связанные с заданной службой (параметр service-name, ftp или wap-wtp), одному и тому же реальному серверу. В модуле CSM допускается спаривать с инициирующим управляющим сеансом только ftp-соединения.

г) Управление доступом к виртуальному серверу (*необязательно*)

- Разрешение на использование виртуального сервера только для определенных клиентов (*необязательно*).

IOS-команда	`client` *ip-address network-mask* (режим конфигурирования виртуального сервера)
CSM-команда	`client` *ip-address network-mask* [`exclude`] (режим конфигурирования виртуального сервера)

Клиентам, имеющим IP-адреса в диапазоне, заданном параметрами *ip address* (стандартно — адрес 0.0.0.0, т.е. все адреса) и *network-mask* (стандартно — 255.255.255.255, т.е. все сети), разрешено подключаться к данному виртуальному серверу. Параметр *network-mask* в этом случае подобен маске списка доступа, в котором первые биты игнорируются, а нулевые сравниваются. В модуле CSM можно использовать ключевое слово exclude, позволяющее запретить указанным адресам доступ к серверу.

- Разрешение на использование виртуального сервера лишь для отправителей из указанной VLAN-сети (*необязательно; только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`vlan` *vlan-number* (режим конфигурирования виртуального сервера)

Стандартно виртуальный сервер принимает соединения, поступающие из любой VLAN-сети. Чтобы ограничить такой доступ, указывается номер VLAN-сети (параметр *vlan-number*: от 2 до 4095), которой разрешается

подключаться к серверу. После определения такой сети всем остальным VLAN доступ к этому виртуальному серверу запрещен.

д) Назначение соединений от клиента к одному и тому же реальному серверу (*необязательно*).

IOS-команда	`stieky` *duration* [`group` *group-id*] [`netmask` *netmask*] (режим конфигурирования виртуального сервера)
CSM-команда	`stieky` *duration*) [`group` *group id*] [`netmask` *netmask*] (режим конфигурирования виртуального сервера)

Соединения заданного клиента назначаются реальному серверу, которой использовался клиентом последним, на время, заданное параметром *duration* (в системе IOS от 0 до 65 535 секунд, в CSM — от 1 до 65 535 секунд). Виртуальным серверам может назначаться идентификатор группы (параметр *group-id*, от 0 до 55, стандартно — 0), связывающий их как отдельную группу. Маска сети (*netmask*, стандартно 255.255.255.255) может быть задана так, что все адреса получателей, соответствующие этой маске, назначаются одному и тому же реальному серверу.

е) Поддержка открытых соединений после разрыва (*необязательно, только в IOS SLB*).

IOS-команда	`delay` *duration* (режим конфигурирования виртуального сервера)
CSM-команда	Нет

После разрыва TCP-соединения функция SLB способна в течение определенного времени (*duration* — от 1 до 600 секунд; стандартно — 10 секунд) поддерживать сеанс соединения. Такой режим может быть полезен в случае, если пакеты поступают не последовательно и соединение перустанавливается до прибытия последнего пакета данных.

ж) Поддержка открытых соединений после истечения таймаута отсутствия активности (*необязательно*).

IOS-команда	`idle` *duration* (режим конфигурирования виртуального сервера)
CSM-команда	`idle` *duration* (режим конфигурирования виртуального сервера)

Когда с помощью функции SLB обнаружено отсутствие пакетов для какого-либо соединения, это соединение на протяжении определенного времени удерживается в открытом состоянии, прежде чем будет отправлено RST-сообщение. Длительность такого периода определяется параметром *duration* (в системе IOS от 10 до 65535, в CSM — от 4 до 65535 секунд; стандартно — 3600 секунд, или 1 час).

з) Предотвращение SYN-лавины, направленной на реальные серверы (*необязательно, только в IOS SLB*).

IOS-команда	`synguard` *syn-count* [*interval*] (режим конфигурирования виртуального сервера)
CSM-команда	Нет

В технологии SLB осуществляется мониторинг количества принятых для виртуального сервера SYN-пакетов. Если количество таких пакетов, полученных в течение интервала времени (параметр `interval` — от 50 до 5000 миллисекунд, стандартно — 100 мс), превышает значение параметра `syn-count` (от 0 до 4294967295; стандартно — 0 или SYN-мониторинг отключен), то все последующие SYN-пакеты удаляются.

и) Управление анонсами виртуального сервера (*необязательно*).

IOS-команда	**advertise [active]** (режим конфигурирования виртуального сервера)
CSM-команда	**advertise [active]** (режим конфигурирования виртуального сервера)

Стандартно при SLB балансировке для адреса виртуального сервера создается статический маршрут на логический интерфейс Null0. Этот маршрут впоследствии может быть переназначен и анонсирован протоколом маршрутизации. Использование ключевого слова **active** приводит к тому, что этот маршрут объявляется только когда доступен по крайней мере один реальный сервер. Объявление такого маршрута можно отключить с помощью ключевой фразы **no advertise**, предотвращающей создание статического маршрута.

к) Установка уровня анализа URL-адресов и cookie-файлов (*необязательно*).

IOS-команда	Нет
CSM-команда	**parse-length** *bytes* (режим конфигурирования виртуального сервера)

Модуль CSM при поиске URL-адресов и информации cookie-файлов анализирует некоторую информацию, количество которой задается параметром *bytes* (от 1 до 4000 байтов, стандартно — 500 байтов).

л) Включение постоянных соединений протокола HTTP 1.1 (*необязательно, только для модуля CSM*).

IOS-команда	Нет
CSM-команда	**persistent rebalance** (режим конфигурирования виртуального сервера)

Стандартно постоянные соединения (persistent connections) для протокола HTTP 1.1 не поддерживаются. Чтобы включить поддержку таких соединений, используется приведенная выше команда.

м) Точная настройка URL-хэширования для балансировки нагрузки (*необязательно; только для модуля CSM*).

IOS-команда	Нет
CSM-команда	**url-hash {begin-pattern | end-pattern}** *pattern* (режим конфигурирования виртуального сервера)

При использовании алгоритма балансировки нагрузки с "прогнозирующим хешем адресов URL" (predictor hash URL) стандартно хешируется весь URL-адрес. Чтобы указать часть URL, подлежащую хешированию, определяется граничный текст `begin-pattern` *pattern* (текстовая строка, с которой начинается хешируемая часть) или `end-pattern` *pattern* (текстовая строка, завершающая хешируемую часть URL). Хешируемая часть включает в себя начальный образец и охватывает строку до завершающего образца. Чтобы определить оба шаблона — начальный и завершающий, указанную команду необходимо применить дважды.

11) Использование SLB-правил для управления трафиком седьмого уровня *(необязательно; только для модуля CSM)*.

IOS-команда	Нет
CSM-команда	`slb-policy` *policy-name* (режим конфигурирования виртуального сервера)

Можно использовать одну или несколько команд `slb-policy` для определения правил, управляющих балансировкой нагрузки, связанной с трафиком вышестоящего уровня. Значение параметра *policy name* (текстовая строка) является именем правила, назначенного на уровне 4,е. В случае, если перечислены несколько правил, они проверяются последовательно. Правило с наивысшим приоритетом должно быть введено первым.

12) Использование функций SLB виртуального сервера.

IOS-команда	`inservice [standby` *group-name*`]` (режим конфигурирования виртуального сервера)
CSM-команда	`inservice` (режим конфигурирования виртуального сервера)

- Стандартно функции SLB не использует виртуальный сервер, если он не подключен. Для отключения виртуального сервера используется ключевая фраза `no inservice`.

Совет

Чтобы обеспечить избыточность для виртуальных серверов, можно использовать несколько SLB-устройств. Функция *некумулятивного резервирования IOS SLB* позволяет каждому SLB-устройству прослушивать HSRP-сообщения от интерфейсов третьего уровня на избыточных коммутаторах. Если один коммутатор (и его IOS-SLB-блок) выходит из строя, основным шлюзом становится другой HSRP-интерфейс. Когда другое SLB-устройство совместно с системой IOS также обнаруживает сбой, активными становятся виртуальные серверы, связанные с данной HSRP-группой *group-name* (определена выше). Однако информация об SLB-состоянии не сохраняется, поэтому существующие соединения разрываются и нужна их повторная установка.

Некумулятивное резервирование требует, чтобы процесс HSRP был сконфигурирован на всех резервных устройствах третьего уровня в сети VLAN *серверной стороны*. Необходимо обеспечить совпадение имени *group-name* в конфигурации протокола HSRP и виртуальных серверов. Более подробные сведения по конфигурированию протокола HSRP приведены в разделе "8.6: резервирование маршрутизаторов с помощью протокола HSRP".

п) Использование кумулятивного резервирования SLB (stateful backup) (*необязательно*).

IOS-команда	`replicate casa` *listening-ip remote-ip port-number [interval]* [`password` [`0` | `7`] *password [timeout]*] (режим конфигурирования виртуального сервера)
CSM-команда	`replicate csrp {sticky | connection}` (режим конфигурирования виртуального сервера)

В операционной системе IOS служба SLB копирует и обменивается своими таблицами решений (decision tables) по распределению нагрузки с другими устройствами кумулятивного резервирования, используя механизм CASA (*Cisco Appliance Services Architecture — структура служб устройств Cisco*). При возникновении сбоя резервное SLB-устройство может немедленно принять на себя выполнение основных SLB-функций.

Эта информация отправляется с адреса прослушивания (параметр *listening-ip* — интерфейс на локальном устройстве) на отдаленный адрес (параметр *remote-ip* — интерфейс на резервном устройстве) с использованием TCP-порта (*port-number* — от 1 до 65 535). Сообщения репликации отправляются с интервалом, заданным параметром *interval* (от 1 до 300; стандартно — 10 секунд).

Для MD5-аутентификации резервного устройства может использоваться пароль (параметр *password* — текстовая строка, 0 используется, если строка не шифруется, 7 — если шифруется). Необязательный параметр *timeout* (от 0 до 65 535 секунд; стандартно — 180 секунд) определяет период времени, в течение которого старый пароль может быть заменен новым. За этот период можно использовать оба пароля — как старый, так и новый.

Модуль CSM реплицирует сведения по своим соединениям с помощью *протокола дублирования параметров коммутации по содержимому* (*Content Switching Replication Protocol — CSRP*). Может производиться либо репликация базы данных `sticky`-соединений, либо репликация базы данных обычных соединений (`connection`). Для репликации обеих баз данных необходимо указать каждую из них в отдельной команде `replicate csrp`.

6. Использование для балансировки нагрузки по серверам *протокола динамической обратной связи* (*Dynamic Feedback Protocol — DFP*) (*необязательно*)

а) Использование DFP-диспетчера для обмена данными между DFP-агентами на серверах (*необязательно*).

- Включение DFP-диспетчера.

IOS-команда	`ip slb dfp` [`password` [`0` | `7`] *password [timeout]*] (режим глобальной конфигурации)
CSM-команда	`dfp` [`password` *password [timeout]*]

DFP-диспетчером балансировки нагрузки может стать маршрутизатор. Для MD5-аутентификации узлового агента в конфигурации протокола DFP может быть задан пароль (*password* — текстовая строка; 0 используется, если строка не шифруется, 7 — если шифруется). Необязательный

параметр `timeout` (от 0 до 65 535 секунд; стандартно — 180 секунд) определяет период времени, в течение которого старый пароль может быть заменен новым. За этот период можно использовать оба пароля — как старый, так и новый.

- Указание DFP-агента.

IOS-команда	**agent** *ip-address port-number* [*timeout* [*retry-count* [*retry-interval*]]] (режим конфигурирования протокола DFP)
CSM-команда	**agent** *ip-address port-number* [*timeout* [*retry-count* [*retry-interval*]]] (режим конфигурирования протокола DFP)

DFP-агент на реальном сервере определяется IP-адресом (*ip-address*) и номером используемого порта (*port-number*). DFP-агент (или сервер) должен контактировать с DFP-диспетчером (IOS SLB-устройством) через заданные промежутки времени (`timeout`, от 0 до 65 535 секунд, стандартно — 0 секунд или без таймаута). DFP-диспетчер совершает множество попыток (их количество определяется значением `retry-count` — от 0 до 65 535; стандартно 0 или бесконечное число попыток) восстановить подключение к агенту с интервалом, равным значению `retry interval` (от 1 до 65 535 секунд; стандартно — 180 секунд).

б) Использование DFP-агента для предоставления DFP-отчетов (*необязательно*).

- Указание агента.

IOS-команда	**ip dfp agent** *subsystem-name* (режим глобальной конфигурации)
CSM-команда	**Нет**

DFP-агент отправляет периодические отчеты своему диспетчеру, главному распределительному устройству (distributed-director). **Имя** подсистемы (*subsystem-name* — текстовая строка длиной до пятнадцати символов) позволяет диспетчеру связывать отчеты сервера с какой-либо подсистемой, управляемой SLB-устройством, для глобальной балансировки нагрузки. Чтобы узнать, какие значения этого параметра доступны с глобального диспетчера, используется команда **ip dfp agent ?**

- Установка пароля для DFP-агента (*необязательно*).

IOS-команда	**password** [0 | 7] *password* [*timeout*] (режим конфигурирования протокола DFP)
CSM-команда	**Нет**

Для MD5-аутентификации DFP-диспетчера можно использовать пароль (параметр *password* — текстовая строка. 0 используется, если строка не шифруется, 7 — если шифруется). Необязательный параметр `timeout` (от 0 до 65 535 секунд; стандартно — 180 секунд) определяет период времени,

в течение которого старый пароль может быть заменен новым. За этот период можно использовать оба пароля — как старый, так и новый.

- Установка номера порта DFP

IOS-команда	`port` *port-number* (режим конфигурирования протокола DFP)
CSM-команда	`manager` *port number* (режим конфигурирования протокола DFP)

DFP-диспетчер и агенты обмениваются данными, используя общий номер порта (параметр *port-number* — от 1 до 65535, стандартного значения нет). DFP-диспетчеры динамически обнаруживают агентов, при этом требуется, чтобы номер порта диспетчера (главного распределительного устройства) был идентичен номеру порта агентов (устройств IOS SLB).

- Установка интервала перерасчета весов (*необязательно*)

IOS-команда	`interval` *seconds* (режим конфигурирования протокола DFP)
CSM-команда	Нет

Прежде чем значения весов DFP-серверов будут предоставлены DFP-диспетчеру, с заданным интервалом времени (параметр *seconds* — от 5 до 65 535 секунд; стандартно — 10 секунд) выполняется их перерасчет.

- Включение DFP-агента.

IOS-команда	`inservice` (режим конфигурирования протокола DFP)
CSM-команда	Нет

Стандартно DFP-агент отключен.

Пример конфигурирования SLB-балансировки

Рассмотрим диаграмму сети на рис. 10.2. Функция SLB сконфигурирована для обеспечения балансировки нагрузки двух серверных групп *FARM1* и *FARM2*.

FARM1 — серверная группа, состоящая из трех реальных web-серверов с IP-адресами 192.168.250.10, 192.168.250.11 и 192.168.250.12. Считается, что реальные серверы находятся в аварийном состоянии, если четыре последовательных попытки установки TCP-соединения с ними терпят неудачу. SLB ожидает 30 секунд, прежде чем предпринимается попытка установить другое соединение с вышедшим из строя сервером. (Количество неудачных TCP-подключений и интервал повторных попыток поддерживаются только в наборе команд операционной системы IOS.) Для проверки соединения с каждым реальным сервером в серверной группе каждые 120 секунд выполняется HTTP-запрос.

В виртуальном сервере *VSERVER1* с адресом 10.10.10.101 для балансировки нагрузки между реальными серверами используется взвешенный алгоритм по минимальному количеству соединений. Новые соединения на 60 секунд переводятся в `sticky`-режим (отправляются к последнему использовавшемуся тем же клиентом серверу).

CSM-версии в этом примере также включает в себя номера VLAN-сетей клиентской и серверной сторон (10 и 20) и IP-адреса (10.10.10.2 и 192.168.250.1).

Одному из серверов назначен вес 32, второму — 16 и третьему — 8. Новые соединения назначаются серверу с наименьшим числом активных соединений, которое определяется емкостью сервера. Например, сервер 192.168.254.10 имеет вес, равный 32, и емкость 32/(32+16+8), или 32/56. Сервер 192.168.254.11 имеет вес 16 и емкость 16/(32+16+8), или 16/56. Сервер 192.168.254.12 имеет вес 8 и емкость 8/(32+16+8), или 8/56. В любой момент времени новое соединение получает сервер, число активных соединений которого наименее приближено к его емкости.

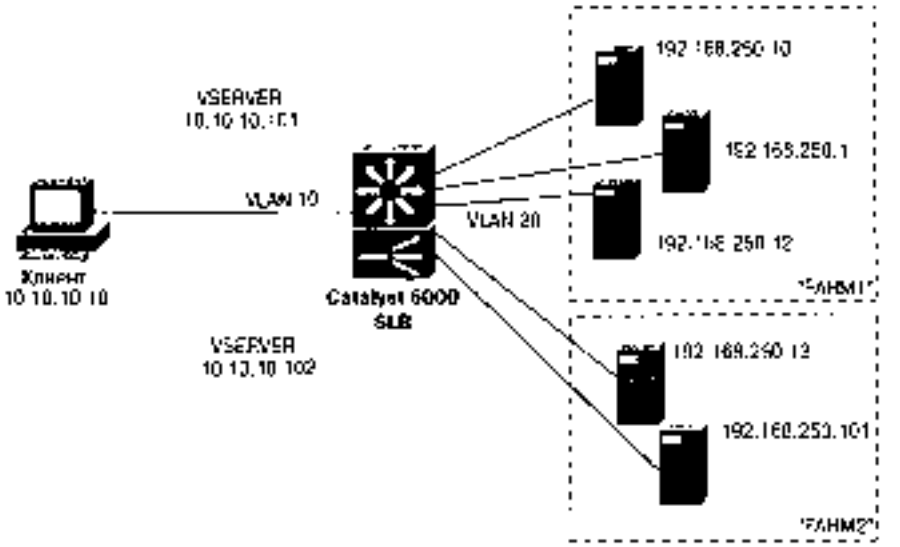


Рис. 10.3. Диаграмма сети для примера SLB-конфигурации

В приведенном ниже конфигурации демонстрируются команды настройки серверной группы FARM1 и виртуального сервера VSERVER1. Приводятся одинаковые конфигурации для IOS-коммутатора и CSM-модуля.

IOS-команды

```
ip slb serverfarm FARM1
(режим глобальной конфигурации, остальные команды вводятся либо в режиме конфигурирования блока серверов, либо в режиме конфигурирования свойств сервера)
predictor leastconns
nat server
probe HTTP1
real 192.168.250.10
weight 32
faildetect numconns 4
retry 30
inservice
exit
real 192.168.250.11
weight 16
faildetect numconns 4
```

```
retry 30
inservice
exit
real 192.168.250.12
weight 8
faildetect numconns 4
retry 30
inservice
exit

ip slb vserver VSERVER1
(режим глобальной конфигурации, остальные команды вводятся либо
в режиме конфигурирования виртуального сервера, либо в режиме на-
стройки протокола DFP, либо в режиме конфигурирования запросов)
serverfarm FARM1
virtual 10.10.10.101 tcp www
sticky 60 group 1
advertise active
inservice
exit
ip slb dfp password 0 test123
agent 192.168.250.10 2000
agent 192.168.250.11 2000
agent 192.168.250.12 2000
exit
probe HTTP1 http
interval 120
port 80
request method get
exit
```

CSM-команды

```
module csm 3
(режим глобальной конфигурации, остальные команды вводятся либо
в режиме конфигурирования виртуального сервера, либо в режиме на-
стройки протокола DFP, либо в режиме конфигурирования запросов)
vlan 10 client
ip address 10.10.10.2 255.255.255.0
gateway 10.10.10.1
exit
vlan 20 server
ip address 192.168.250.1 255.255.255.0
exit
serverfarm FARM1
predictor leastconns
nat server
probe HTTP1
real 192.168.250.10
weight 32
inservice
exit
```

```
real 192.168.250.11
weight 16
inservice
exit
real 192.168.250.12
weight 8
exit
vserver VSERVER1
serverfarm FARM1
virtual 10.10.10.101 tcp www
sticky 60 group 1
advertise active
inservice
exit
dfp password test123
agent 192.168.250.10 2000
agent 192.168.250.11 2000
agent 192.168.250.12 2000
exit
probe HTTP1 http
interval 120
request method get
exit
```

Вторая серверная группа FARM2 состоит из двух реальных серверов с адресами 192.168.250.12 и 192.168.250.101. IOS SLB-устройство сконфигурировано на стандартную балансировку нагрузки с циклическим алгоритмом. Вместе с тем модуль CSM предоставляет больше возможностей. В нашем случае используется алгоритм балансировки нагрузки с хэшированием URL-адресов. Каждые 60 секунд отсылаются GET-запросы протокола HTTP2 на реальных серверах. Если пять запросов терпят неудачу, сервер переводится в аварийное состояние.

Виртуальному серверу VSERVER2 назначен адрес 10.10.10.102 для HTTP-трафика. В модуле CSM для этого виртуального сервера сконфигурированы два правила балансировки нагрузки. В правиле 1 используется таблица URL1, предназначенная для отбора URL-адресов, в которых содержится подстрока "/signup/*" либо "/signout/*". Кроме того, проверяются "залипшие" SSL-соединения. В правиле 2 используется таблица проверки cookie-файлов Cart, предназначенная для поиска файла с именем "myname", содержащего значение "Sloga*". Кроме того, в стратегии 2 также проверяются "липучие" соединения, в которых имеются cookie-файлы с именем "test".

В приведенной ниже конфигурации демонстрируются команды, необходимые для настройки серверной группы FARM2 и виртуального сервера VSERVER2.

IOS-команды

```
ip slb serverfarm FARM2
(режим глобальной конфигурации, остальные команды вводятся либо в режиме конфигурирования блока серверов, либо в режиме конфигурирования свойств сервера)
predictor roundrobin
nat server
probe HTTP2
real 192.168.250.11
```

	inservice exit real 192.168.250.101 inservice exit ip slb vserver VSERVER2 (режим глобальной конфигурации, остальные команды вводятся либо в режиме конфигурирования виртуального сервера, либо в режиме сброса) serverfarm FARM2 virtual 10.10.10.102 tcp 80 inservice exit probe HTTP2 http credentials testuser test123 request method get /home interval 60 faildetect 5
CSM-команды	serverfarm FARM2 predictor hash url nat server probe HTTP2 real 192.168.250.13 inservice exit real 192.168.250.101 inservice exit vserver VSERVER2 serverfarm FARM2 virtual 10.10.10.102 tcp 80 slb-policy Policy1 slb-policy Policy2 inservice exit map Cart cookie match protocol http cookie mycoke cookie-value &bop* exit map URL1 url match protocol http url /signUp/* match protocol http url /support/* exit sticky 1 ssl sticky 2 cookie test policy Policy1 url-map URL1 sticky-group 1 serverfarm FARM2 exit policy Policy2

```
cookie-map Cart
sticky-group 2
serverfarm FARM2
exit
probe HTTP2 http
credentials testuser test123
request method get /home
interval 50
retries 5
exit
```

Отображение сведений, касающихся SLB-балансировки нагрузки

В табл. 10.1 перечислены некоторые команды коммутатора, которые можно использовать для отображения полезной информации по SLB-конфигурации и состоянию службы балансировки нагрузки.

Таблица 10.1. Команды для отображения информации о функции балансировки нагрузки на серверы и ее состоянии

Функция отображения	Операционная коммутатора	Команда
VLAN-назначение	IOS	Нет
	CSM	`show module csm slot vlan [client |server ft] [id vlan-id] [detail]`
Серверные группы	IOS	`show ip slb serverfarms [name serverfarm-name] [detail]`
	CSM	`show module csm slot serverfarms [name serverfarm-name] [detail]`
Реальные серверы	IOS	`show ip slb reals [vserver virtual-server-name] [detail]`
	CSM	`show module csm slot real [sfarm sfarm name] [detail]`
CSM правила	IOS	Нет
	CSM	`show module csm slot policy [name policy-name]`
Виртуальные серверы	IOS	`show ip slb vserver [name virtual-server-name] [detail]`
	CSM	`show module csm slot vserver [detail]`
Виртуальные серверы перенаправления	IOS	Нет
	CSM	`show module csm slot vserver redirect`
SLB-соединения	IOS	`show ip slb conns [vserver virtual-server-name , client ip-address] [detail]`

Функция отображения	Операционная коммутатора	Команда
	CSM	`show module csm slot conns [vserver virtualserver- name] [client ip-address] [detail]`
Состояние протокола DFP	IOS	`show ip slb dfp [agent agent-ip address port-number | manager manager-ip-address , detail | weights]`
	CSM	`show module csm slot dfp [agent [detail | ip-address port] manager [ip-addr] | detail | weights]`
SLB-резервирование	IOS	`show ip slb replicate`
	CSM	`show module csm slot ft [detail]`
Запросы	IOS	`show ip slb probe [name probe_name] [detail]`
	CSM	`show module csm slot probe [http | icmp , telnet | tcp | ftp | smtp | dns] [name probe_name] [detail]`
SLB-статистика	IOS	`show ip slb stats`
	CSM	`show module csm slot stats`

10.2: балансировка нагрузки на брандмауэры

- В процессе балансировки нагрузки на брандмауэры распределяются потоки данных, направленные к одной или нескольким группам брандмауэров.
- Группа брандмауэров (firewall farm) — это группа устройств, которые подключены параллельно или имеют "внутренние" (защищенные) и "внешние" (незащищенные) интерфейсы, подключенные к общим сетевым сегментам.
- Для балансировки нагрузки на брандмауэры необходимо подключить устройство балансировки (устройство с запущенной службой SLB под управлением операционной системы IOS) к обеим сторонам группы брандмауэров. Группе брандмауэров с "внутренними" и "внешними" интерфейсами требуется, кроме того, два балансирующих устройства. Каждое такое устройство позволяет направлять потоки данных одному и тому же брандмауэру в течение периода соединения. Основная схема балансировки нагрузки на брандмауэры показана на рис. 10.3.
- Балансировка нагрузки на брандмауэры осуществляется путем вычисления хэш-значения каждого входного потока данных (IP-адреса и порты отправителя и получателя).
- Устройством балансировки маскируется под единым IP-адресом для всех брандмауэров в группе.
- Посредством мониторинга запросов об активности при балансировке нагрузки может быть обнаружен отказ какого-либо брандмауэра.

- Протокол **HSRP** может применяться для обеспечения "некумулятивного резервирования" для множества устройств балансировки нагрузки. Если в одном из устройств возникает сбой, то его функции может принять на себя запасное устройство.

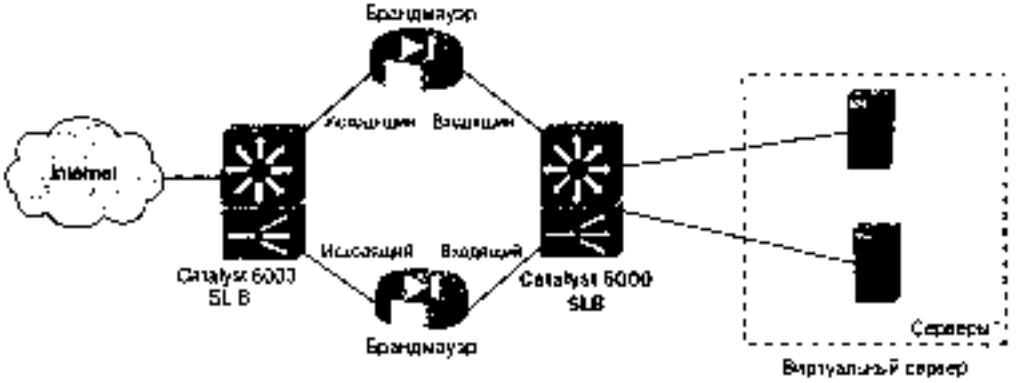


Рис. 10.3. Схема балансировки нагрузки по брандмауэрам

- Множество устройств балансировки нагрузки для создания механизма резервирования может также использовать "кумулятивное резервирование". Резервные устройства динамически сохраняют конфигурацию о состоянии и могут вступить в действие немедленно после возникновения сбоя.

Конфигурирование службы

1. Указание VLAN-сетей клиентской и серверной стороны (*только для модуля CSM*).

   а) Начало конфигурирования CSM-модуля.

IOS-команда	Нет
CSM-команда	`module` *slot_number* (режим глобальной конфигурации)

В интерфейсе командной строки операционной системы IOS запускается режим конфигурации модуля CSM, расположенного в гнезде с номером *slot_number* шасси коммутатора. Для выхода из этого режима используется команда `exit`. Чтобы определить номер соответствующего гнезда, применяется команда `show module all`.

   б) Указание типов всех VLAN-сетей.

IOS-команда	Нет
CSM-команда	`vlan` *vlan-id* {`client` | `server`} (режим конфигурирования модуля CSM)

Номер VLAN-сети задается значением *vlan-id* (2-4095; использование сети VLAN с номером 1 невозможно). Эта сеть уже должна быть определена в базе данных VLAN-сетей коммутатора. Тип VLAN-сети — `client` или `server` — определяет, где с точки зрения модуля CSM размещены клиенты и серверы. Прежде

чем можно будет соответствующим образом использовать модуль CSM, необходимо определить сети VLAN обоих типов: как клиентские, так и серверные.

Совет

В модуле CSM не предусмотрен специальный конфигурационный режим для групп брандмауэров или балансировки нагрузки на брандмауэры. Его заменяет рассмотренная ранее концепция балансировки нагрузки с клиентской и серверной сторон. При конфигурировании модуля CSM для балансировки нагрузки на брандмауэры всегда следует учитывать то, что сеть VLAN клиентской стороны модуля должна быть удалена от брандмауэров. Ближайшей или непосредственно подключенной к группам брандмауэров всегда является VLAN-сеть серверной стороны модуля CSM.

в) Назначение основного IP-адреса (*необязательно*).

IOS-команда	Нет
CSM-команда	`ip address` *ip-address netmask*

В модуле CSM для каждой сети VLAN может быть определен один IP-адрес. Он используется для административного трафика (например, проб) и ARP-запросов.

г) Назначение дополнительного вторичного IP-адреса (*необязательно*).

IOS-команда	Нет
CSM-команда	`alias` *ip address netmask*

Дополнительный IP-адрес позволяет модулю CSM обмениваться пакетами с серверами в другой сети без маршрутизатора.

д) Выбор стандартного шлюза (*необязательно*).

IOS-команда	Нет
CSM-команда	`gateway` *ip-address*

Адрес стандартного шлюза следующего транзитного перехода или маршрутизатора задается значением *ip-address*. Эту команду можно повторить для определения до семи шлюзов в каждой VLAN-сети или 255 шлюзов в модуле CSM. Шлюзы обычно используются во VLAN-сети клиентской стороны, хотя, если потребуется, могут использоваться и на серверной стороне.

е) Указание статических маршрутов для связи с удаленными сетями (*необязательно*).

IOS-команда	Нет
CSM-команда	`route` *ip-address netmask* `gateway` *gw-ip-address*

Статический маршрут может быть задан тогда, когда CSM-модулю необходимо получать сведения о том, как достичь серверов, расположенных на расстоянии более одного транзитного перехода. Маршрут определяется параметрами *ip-address* и *netmask* с использованием адреса шлюза *gw-ip-address*. Шлюз должен быть расположен в той же локальной сети, что и CSM VLAN.

ж) Повторение этапов б)-е) для каждой VLAN-сети клиентской и серверной стороны.

2. Указание группы брандмауэров

а) Назначение имени группе брандмауэров.

IOS-команда	`ip slb firewallfarm` *firewallfarm-name* (режим глобальной конфигурации)
CSM-команда	`serverfarm` *serverfarm-name* (режим конфигурирования модуля CSM)

В операционной системе IOS SLB-ссылка на набор брандмауэров задается значением *firewallfarm-name* (текстовая строка длиной до пятнадцати символов). В то же время модуль CSM рассматривает группу брандмауэров как фермы серверной группы, поэтому ссылка задается значением *serverfarm-name* (текстовая строка длиной до пятнадцати символов).

Совет

Помните: для балансировки нагрузки на брандмауэры требуются два устройства балансировки — по одному на каждой стороне группы брандмауэров. В операционной системе IOS SLB группа брандмауэров конфигурируется с помощью параметра `firewallfarm` как группа, состоящая из реальных серверов или брандмауэров. В результате балансировка нагрузки основывается на "маршрутах" брандмауэров, сконфигурированных в каждом устройстве балансировки. В SLB-методиках отсутствует понятие виртуальных серверов с балансировкой нагрузки на брандмауэры.

Однако в модуле CSM балансировка нагрузки на брандмауэры рассматривается как расширение обычного метода балансировки нагрузки. Следует конфигурировать группу брандмауэров как серверную группу реальных серверов `serverfarm`. Виртуальные серверы конфигурируются для балансировки трафика, направленного к группе брандмауэров. Трафик, исходящий из группы брандмауэров, также необходимо конфигурировать для балансировки нагрузки с общей серверной группой `serverfarm`, не имеющей реальных серверов. Вместо этого общий реальный сервер пересылает весь трафик через обычную серверную группу согласно внутренним таблицам маршрутизации модуля CSM.

б) Идентификация одного или нескольких брандмауэров в группе.

- Указание IP-адреса брандмауэра.

IOS-команда	`real` *ip-address* (режим конфигурирования группы брандмауэров)
CSM-команда	`real` *ip-address* (режим конфигурирования серверной группы)

Брандмауэр непосредственно (в той же логической подсети) подключается к устройству балансировки нагрузки с помощью интерфейса с IP-адресом *ip-address*.

- Назначение весового коэффициента устройства (*необязательно; подробнее в IOS SLB*).

IOS-команда	`weight` *weighting value* (режим конфигурирования свойств брандмауэра)

CSM-команда	Нет

Реальному брандмауэру назначается весовой коэффициент (параметр `weighting value` — от 1 до 255; стандартно — 8), который указывает "емкость" (т.е. производительность) устройства по отношению к другим реальным брандмауэрам в группе. Эти значения определяются статически и основываются на предполагаемой нагрузке данного брандмауэра по отношению к другим брандмауэрам. Значения веса используются только для циклического алгоритма или алгоритма, учитывающего наименьшее число соединений.

- Указание одного или нескольких запросов для обнаружения отказа брандмауэра (*необязательно; только для службы SLB операционной системы IOS*).

IOS-команда	`probe` *probe-name* (режим конфигурирования свойств брандмауэра)
CSM-команда	Нет

Запрос, заданный с помощью имени (параметр *probe name* — текстовая строка), периодически используется для определения работоспособности брандмауэра. В случае, если один из тестов оказывается неудачным, брандмауэр объявляется вышедшим из строя, даже если определено несколько тестов. Для восстановления брандмауэру необходимо пройти все тесты.

Совет

Помимо указанных выше действий, необходимо определять тесты по отдельности, как описывается в разделе "10.3: SLB-тесты". Для балансировки нагрузки на брандмауэры наиболее полезными являются ping-тесты. Для каждого брандмауэра в группе следует настроить какой-либо тест, предназначенный для отправки ping-пакетов, полностью проходящих через брандмауэр и направленных устройству балансировки нагрузки на удаленном конце. В таком случае тестируются как "внутренние", так и "внешние" интерфейсы брандмауэра, вынужденные активизироваться и выполнять функции так, чтобы ping-тест был отражен с другой стороны. Необходимо обеспечить такие настройки брандмауэра, которые позволят ping-пакетам протокола ICMP проходить через него.

- Разрешение на использование брандмауэра функцией балансировки нагрузки.

IOS-команда	`inservice` (режим конфигурирования свойств брандмауэра)
CSM-команда	`inservice` (режим конфигурирования свойств сервера)

Стандартно реальный брандмауэр не используется в службе SLB, если он не подключен к обслуживанию. Для отключения брандмауэра от обслуживания используется команда `no inservice`.

в) Указание одного или нескольких потоков, которые будут отправляться данной группе брандмауэров (*необязательно; только для службы SLB операционной системы IOS*).

IOS-команда	**access [source** *source-ip-address network-mask*] **[destination** *destination-ip-address network-mask*] (режим конфигурирования свойств группы брандмауэров)
CSM-команда	Нет

Если существуют несколько групп брандмауэров, трафик можно идентифицировать по адресу и отправлять через соответствующую группу брандмауэров. Поток данных определяется адресами отправителя и получателя, а также масками их подсетей. Если в приведенной выше команде ключевые слова **source** и **destination** опущены, то стандартно адрес 0.0.0.0 с маской 0.0.0.0 — все адреса и сети. Такая установка является стандартной.

При использовании модуля CSM балансировка трафика осуществляется в группе брандмауэров с помощью виртуального сервера и алгоритмов балансировки нагрузки.

г) Выбор метода балансировки нагрузки на брандмауэры (*необязательно*)

IOS-команда	**predictor hash address [port]** (режим конфигурирования группы брандмауэров)
CSM-команда	**predictor hash address source 255.255.255.255** или **predictor hash address destination 255.255.255.255** (обе команды вводятся в режиме конфигурирования серверной группы)

Стандартно в службе SLB операционной системы IOS для выбора брандмауэра получателя используются IP-адреса отправителя и получателя потока. Чтобы при выборе брандмауэра учитывались адреса отправителя и получателя, а также номера их TCP- или UDP-портов, необходимо использовать ключевое слово **port**.

В модуле CSM, расположенном с "внешней" (небезопасной) по отношению к группе брандмауэров стороны, в алгоритме следует использовать только адреса отправителей (**source**). В данном случае маска сети равна **255.255.255.255** для того, чтобы в алгоритме хэширования использовались все биты адреса. В CSM-модуле, расположенном с "внутренней" (безопасной) по отношению к группе брандмауэров стороны, следует использовать адреса получателей (**destination**) и маску **255.255.255.255**. Алгоритм хэширования обеспечивает наилучшее распределение нагрузки, если основывается на большом количестве адресов, расположенных вдали от группы брандмауэров. (Здесь предполагается, что размер сети и количество узлов и IP-адресов увеличивается при удалении от группы брандмауэров.)

д) Отключение NAT-трансляции адресов сервера (*только для модуля CSM*).

IOS-команда	Нет
CSM-команда	**no nat server** (режим конфигурирования серверной группы)

Стандартно NAT-преобразование адреса сервера включено для серверной группы CSM. При балансировке нагрузки на брандмауэры группа брандмауэров рассматривается как серверная группа. В то же время при балансировке нагрузки на брандмауэры NAT-преобразование не является необходимым.

е) Указание одного или нескольких тестов для обнаружения сбоев внутри группы брандмауэров (*необязательно, только для модуля CSM*).

IOS команда	Нет
CSM-команда	**probe** *probe-name* (режим конфигурирования серверной группы)

Посредством заданного по имени (параметр *probe-name* – текстовая строка) теста периодически проверяется каждый брандмауэр (реальный сервер) внутри группы брандмауэров. В качестве целевого адреса тест получает IP-адрес каждого реального сервера. Если один из запросов оказывается неудачным, брандмауэр объявляется выпавшим из строя, даже если определены несколько тестов. Для восстановления брандмауэру необходимо пройти все тесты.

Совет

Задавать тесты нужно по отдельности, как описано в разделе "10.3: SLB-тесты". Наиболее полезными для балансировки нагрузки на брандмауэры являются **ping**-тесты.

В отличие от службы SLB операционной системы IOS, при балансировке нагрузки на брандмауэры модуль CSM только предоставляет тесты, которые можно использовать в целях на серверной группе. Иными словами, невозможно определить тест с уникальными целевыми адресами. В идеальном случае тест следует конфигурировать так, чтобы **ping**-пакет, адресованный устройству балансировки нагрузки на удаленной стороне, полностью проходил через брандмауэр. В таком случае тестируются как "внутренние", так и "внешние" интерфейсы брандмауэра, что вынуждает их быть активными и выполнять функции таким образом, чтобы **ping**-тест был отражен с другой стороны.

CSM-тесты способны получать целевые адреса только от реальных серверов, определенных в серверной группе. Обычно им является ближайший интерфейс брандмауэра, поэтому из теста можно определить его полную работоспособность.

ж) Использование кумулятивного резервирования для восстановления после сбоя (*необязательно*).

IOS-команда	**replicate case** *listening-ip remote-ip port-number* [*interval*] [**password** {0 | 7} *password* [*timeout*]]
	(режим конфигурирования серверной группы)
CSM-команда	Нет

Резервные устройства балансировки нагрузки для обмена и репликации сведений о состоянии используют CASA-структуру. Эта информация отправляется с адреса *listening-ip* (интерфейс на локальном устройстве) на адрес *remote-ip* (интерфейс резервного устройства) с использованием порта номер *port-number* (1–65535). Сообщения репликации отправляются с интервалом *interval* (от 1 до 300 секунд; стандартно – 10 секунд).

Для MD5-аутентификации резервного устройства может применяться пароль (параметр *password* — текстовая строка; 0 используется, если строка не шифруется, 7 — если шифруется). Необязательный параметр *timeout* (от 0 до 65 535 секунд; стандартно — 180 секунд) определяет период времени, в течение которого старый пароль может быть заменен новым. За этот период можно использовать оба пароля — как старый, так и новый.

7) Регулировка параметров TCP и UDP соединений (*необязательно; только для службы SLB операционной системы IOS*).

- Переход в режим конфигурирования протоколов TCP и UDP.

IOS-команда	`{tcp | udp}` (режим конфигурирования серверной группы)
CSM-команда	Нет

Нередко возникает необходимость настраивать оба протокола. В таком случае можно повторить заданную команду для каждого протокола.

- Поддержка открытых соединений после разрыва (*необязательно; только для протокола TCP*).

IOS-команда	`delay` *duration* (режим конфигурирования серверной группы)
CSM-команда	Нет

После разрыва TCP-соединения это канал может поддерживаться в течение заданного интервала времени (параметр *duration* — 1–600 секунд; стандартно — 10 секунд). Такой режим полезен, если пакеты прибывают непоследовательно и соединение переустанавливается до прибытия последнего пакета данных.

- Поддержка открытых соединений после истечения таймаута отсутствия активности (*необязательно*).

IOS-команда	`idle` *duration* (режим конфигурирования серверной группы)
CSM-команда	Нет

Если в соединении отсутствуют пакеты, соединение поддерживается в открытом состоянии в течение заданного времени (параметр *duration* — от 10 до 65 535 секунд; стандартно — 3600 секунд, или 1 час), прежде чем будет отправлено RST сообщение.

- Конфигурирование максимального количества соединений (*необязательно*).

IOS-команда	`maxconns` *number* (режим конфигурирования серверной группы)
CSM-команда	Нет

В любой момент времени реальный сервер ограничен определенным количеством активных соединений (параметр *number* — 1–4 294 967 295; стандартное значение равно 4 294 967 295).

- Назначение подключения соединений с одного IP-адреса к одному и тому же брандмауэру (*необязательно*).

IOS-команда	`sticky` *duration* [`netmask` *netmask*] (режим конфигурирования серверной группы)
CSM-команда	Нет

Соединения с заданного IP-адреса на определенное время (параметр *duration* — 0–65 535 секунд) назначаются последнему использованному брандмауэру. Маска сети (*netmask*) может быть задана так, что все адреса отправителей, охваченные ею, назначаются одному и тому же брандмауэру.

и) Использование брандмауэра функцией балансировки нагрузки (*только для службы SLB операционной системы IOS*).

IOS-команда	`inservice` (режим конфигурирования серверной группы)
CSM-команда	Нет

Стандартно брандмауэр, если он не подключен к обслуживанию, не используется для балансировки нагрузки на брандмауэры. Для отключения брандмауэра используется ключевой префикс `no` перед `inservice`. Серверная группа модуля CSM, естественно, подключена к обслуживанию.

3. Указание виртуального сервера для обработки трафика, направленного к серверной группе (*только для модуля CSM*).

а) Назначение имени виртуальному серверу

IOS-команда	Нет
CSM-команда	`vserver` *virtual-server-name*

Виртуальному серверу назначается определенное имя (параметр *virtual-server-name* — текстовая строка длиной до пятнадцати символов).

б) Назначение виртуального сервера серверной группе брандмауэра.

IOS-команда	`serverfarm` *serverfarm-name* (режим конфигурирования виртуального сервера)
CSM-команда	Нет

В функции SLB виртуальный сервер используется как клиентская часть серверной группы с именем *serverfarm-name* (текстовая строка длиной до пятнадцати символов).

в) Конфигурирование возможностей виртуального сервера

IOS-команда	Нет
CSM-команда	`virtual` *ip-address* [*network-mask*] *any* (режим конфигурирования виртуального сервера)

Виртуальный сервер представлен IP-адресом *ip-address* (стандартно 0.0.0.0) с маской сети *network-mask* (стандартно 255.255.255.255; бит, равный еди-

нице. совпадает, нулевой бит является инвертированной маской). Группы брандмауэров ip-address и network-mask можно настроить как параметры всей внутренней сети серверов. Такой подход позволяет виртуальным серверам представлять множество реальных машин при одновременной балансировке трафика к брандмауэрам (реальным серверам). Ключевое слово any позволяет осуществлять балансировку нагрузки для всех протоколов.

г) Разрешение на использование виртуального сервера только для трафика VLAN-сети отправителя (*необязательно*).

IOS-команда	Нет
CSM-команда	`vlan` *vlan-number* (режим конфигурирования виртуального сервера)

Стандартно трафик от всех VLAN-сетей может посредством виртуального сервера достигать брандмауэров. Чтобы ограничить эту возможность, следует указать номер VLAN-сети (параметр *vlan-number* — 2–4095), которой разрешается доступ. Обычно такой сетью является VLAN сеть модуля CSM, которая подключена к "внешней" сети, более всего удаленной от брандмауэра. После того как сеть задана, остальным VLAN-сетям доступ к виртуальному серверу запрещен.

д) Разрешение использования виртуального сервера для SLB-балансировки нагрузки.

IOS-команда	Нет
CSM-команда	`inservice` (режим конфигурирования виртуального сервера)

Стандартно виртуальный сервер, если он не подключен к обслуживанию, в методике SLB не используется. Для отключения виртуального сервера используется ключевая фраза `no inservice`.

е) Использование кумулятивного реплицирования SLB (*необязательно*).

IOS-команда	Нет
CSM-команда	`replicate csrp {sticky | connection}` (режим конфигурирования виртуального сервера)

Модуль CSM реплицирует информацию о соединениях посредством протокола CSRP. Можно реплицировать либо базу данных sticky-соединений, либо базу данных обычных соединений (`connection`). Для репликации обеих баз данных каждую из них необходимо указать в отдельной команде `replicate csrp`.

4. Указание общей серверной группы для трафика, исходящего от группы брандмауэров (*только для модуля CSM*).

а) Назначение имени общей серверной группе.

IOS-команда	Нет
CSM-команда	`serverfarm` *serverfarm-name*

Модуль CSM рассматривает сеть, отдаленную от группы брандмауэров, как серверную группу. В модуле CSM, расположенном на "внешней" стороне группы брандмауэров, такой сетью обычно является Internet или общедоступная сеть.

В модуле CSM с "внутренней" стороны группы брандмауэров такой сетью может быть другая внутренняя сеть или действительная серверная группа.

Совет

Трафик, исходящий из группы брандмауэров, также необходимо сконфигурировать для балансировки нагрузки с помощью общей серверной группы `servertarm`, не имеющей реальных серверов. Вместо этого общий виртуальный сервер перенаправит весь трафик через общую серверную группу согласно внутренним таблицам маршрутизации модуля CSM.

б) Выбор метода балансировки нагрузки (*необязательно*).

IOS-команда	Нет
CSM-команда	`predictor forward` (режим конфигурирования серверной группы)

В модуле CSM, расположенном на "внешней" (небезопасной) по отношению к группе брандмауэров стороне, в алгоритме балансировки нагрузки следует использовать только режим `forward`. При использовании этого режима трафик, адресованный за пределы группы брандмауэров, перенаправляется согласно внутренней таблице маршрутизации модуля CSM.

В модуле CSM, расположенном с "внутренней" (безопасной) по отношению к группе брандмауэров стороны, трафик обычно адресуется внутренней сети или действительным группе серверов. В данном случае может использоваться нормальный режим балансировки нагрузки на серверы.

в) Отключение серверного NAT-преобразования

IOS-команда	Нет
CSM-команда	`no nat server` (режим конфигурирования серверной группы)

Стандартно серверное NAT-преобразование включено для серверной группы модуля CSM. При такой балансировке нагрузки на брандмауэры группа брандмауэров рассматривается как серверная группа. Однако NAT-преобразование для балансировки нагрузки на брандмауэры не является необходимым.

5. Указание общего виртуального сервера для обработки трафика, направленного из группы брандмауэров (*только для модуля CSM*).

а) Имя виртуального сервера.

IOS-команда	Нет
CSM-команда	`vserver` *virtual-server-name*

Виртуальному серверу присваивается определенное имя (параметр *virtual-server-name* — текстовая строка длиной до пятнадцати символов).

б) Назначение виртуального сервера серверной группе брандмауэра.

IOS-команда	Нет
CSM-команда	`serverfarm` *serverfarm-name* (режим конфигурирования виртуального сервера)

В четвертке SLB данный виртуальный сервер используется в качестве клиентской части общей серверной группы с именем *serverfarm-name* (текстовая строка длиной до пятнадцати символов).

в) Конфигурирование возможностей виртуального сервера.

IOS-команда	Нет
CSM-команда	`virtual` *ip-address [network-mask]* `any` (режим конфигурирования серверной группы)

Виртуальный сервер представлен IP-адресом *ip-address* (стандартно — 0.0.0.0) с маской сети *network-mask* (стандартно — 255.255.255.255, бит, равный единице, совпадает, нулевой бит является интерпретированной маской). Для представления внешней открытой сети (например, Internet) используется команда `virtual 0.0.0.0 0.0.0.0 any`.

г) Разрешение на использование данного виртуального сервера сетью VLAN определенной (*необязательно*)

IOS-команда	Нет
CSM-команда	`vlan` *vlan-number* (режим конфигурирования серверной группы)

Стандартно достигать брандмауэра пересылкой виртуального сервера может трафик всех VLAN-сетей. Чтобы ограничить доступ, следует указать номер VLAN-сети (*vlan-number* — 2-4095), трафик которой будет разрешен. Как правило, такой сетью является VLAN-сеть модуля CSM, подключенная к группе брандмауэров. После определения сети доступ к виртуальному серверу для всех остальных VLAN-сетей запрещен.

д) Разрешение на использование функцией SLB данного виртуального сервера.

IOS-команда	Нет
CSM-команда	`inservice` (режим конфигурирования серверной группы)

Стандартно виртуальный сервер не используется в службе SLB, если он не подключен к обслуживанию. Чтобы отключить виртуальный сервер от обслуживания, используется команда `no inservice`.

Пример конфигурирования балансировки нагрузки на брандмауэры

Для балансировки нагрузки на брандмауэры необходимы два устройства: одно, расположенное с внешней стороны, и другое — внутри по отношению к группе брандмауэров. На рис. 10.4 приведена диаграмма сети для примера конфигурации.

Группа брандмауэров состоит из двух реальных брандмауэров. Их "внешние" (незащищенные) интерфейсы имеют адреса 192.168.1.2 и 192.168.1.3. Адреса "внутренних" (защищенных) интерфейсов соответственно — 192.168.100.2 и 192.168.100.3. С внешней стороны стандартный шлюз имеет адрес 10.5.1.1, а адрес внешнего устройства балансировки нагрузки — 10.5.1.2.

Внутреннее SLB-устройство осуществляет балансировку нагрузки на брандмауэры для внешнего трафика в группе брандмауэров. Кроме того, это устройство обеспечивает обычную балансировку нагрузки на серверы для внутренней серверной группы. Адреса реальных серверов — 10.70.1.10 и 10.70.1.20, виртуальный сервер представлен адресом 10.5.1.80.

Функционирование брандмауэров тестируется с помощью ping-запросов, осуществляемых как внешним, так и внутренним SLB-устройствами. Каждый реальный сервер данной серверной группы тестируется с помощью HTTP-запросов, которые отправляются каждые 240 секунд стандартным методом GET.

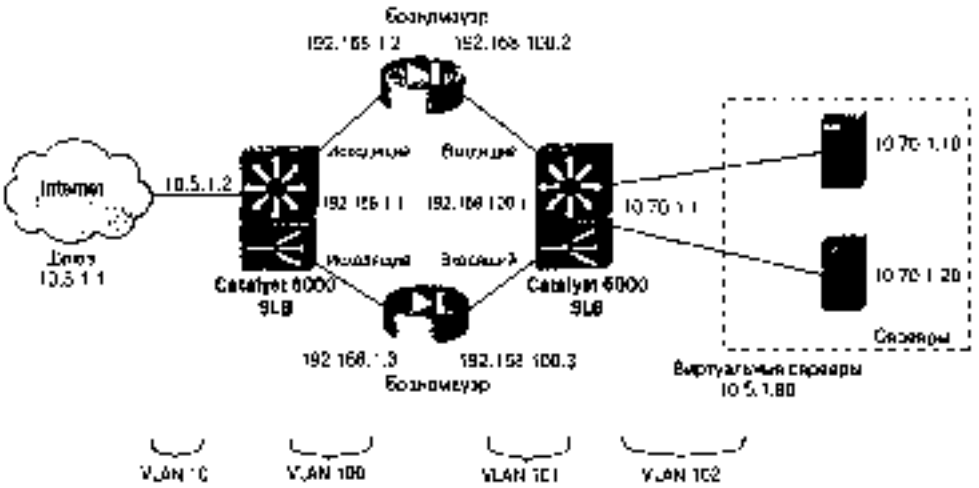


Рис. 10.4. Диаграмма сети к примеру конфигурирования балансировки нагрузки на брандмауэры

Ниже приведена конфигурация для внешнего устройства балансировки нагрузки.

IOS-команды

```
ip slb firewallfarm Outside
real 192.168.1.2
weight 6
probe Ping1
inservice
exit
real 192.168.1.3
weight 8
probe Ping2
inservice
exit
inservice
exit
ip slb probe Ping1 ping
address 192.168.100.1
interval 10
faildetect 4
ip slb probe Ping2 ping
address 192.168.100.1
interval 10
faildetect 4
exit
```

CSM-команды

```
module csm 3
vlan 10 client
ip address 10.5.1.2 255.255.255.0
gateway 10.5.1.1
exit
vlan 100 server
ip address 192.168.1.1 255.255.255.0
exit
serverfarm Outside
real 192.168.1.2
inservice
exit
real 192.168.1.3
inservice
exit
predictor hash address source 255.255.255.255
no nat server
probe Ping1
exit
vserver Voutside
serverfarm Outside
virtual 10.5.1.0 255.255.255.0 any
vlan 10
inservice
exit
serverfarm Internet
predictor forward
no nat server
exit
vserver Vinternet
serverfarm Internet
virtual 0.0.0.0 0.0.0.0 any
vlan 100
inservice
exit
probe Ping1 ping
address 192.168.1.1
retries 4
exit
```

Конфигурация для внутреннего устройства балансировки нагрузки приведена ниже.

IOS-команды

```
ip slb firewallfarm Inside
real 192.168.100.2
weight 8
probe Ping1
inservice
exit
real 192.168.100.3
weight 8
probe Ping2
inservice
exit
```

```
inservice
exit
ip slb serverfarm Servers
nat server
probe HTTP1
real 10.70.1.10
inservice
exit
real 10.70.1.20
inservice
exit
ip slb vserver Vservers
serverfarm Servers
virtual 10.5.1.80 tcp 0
inservice
exit
ip slb probe Ping1 ping
address 192.168.1.1
interval 10
faildetect 4
exit
ip slb probe Ping2 ping
address 192.168.1.1
interval 20
faildetect 4
exit
ip slb probe HTTP1 http
port 80
interval 240
request
```

CSM конфигурация

```
module csm 3
vlan 102 client
ip address 10.70.1.1 255.255.255.0
exit
vlan 201 server
ip address 192.168.100.1 255.255.255.0
exit
serverfarm Inside
real 192.168.100.2
inservice
exit
real 192.168.100.3
inservice
exit
predictor hash address destination 255.255.255.255
no nat server
probe Ping1
exit
vserver Vinside
serverfarm Inside
virtual 0.0.0.0 0.0.0.0 any
vlan 102
```

```
inservice
exit
serverfarm Servers
real 10.70.1.10
inservice
exit
real 10.70.1.20
inservice
exit
probe HTTP1
exit
vserver Vservers
serverfarm Servers
virtual 10.5.1.80 tcp 0
vlan 101
inservice
exit
probe Ping1 ping
address 192.168.1.1
retries 4
exit
probe HTTP1 http
interval 240
request
```

Отображение информации о балансировке нагрузки на брандмауэры

В табл. 10.2 перечислены некоторые команды коммутатора, которые можно использовать для получения полезных сведений о конфигурации и состоянии службы SLB-балансировки нагрузки на брандмауэры.

Таблица 10.2. Команды для отображения сведений о конфигурации и состоянии функции балансировки нагрузки на брандмауэры

Функция отображения	Операционная система коммутатора	Команда
Состояние брандмауэров в группе	IOS	`show ip slb reals`
	CSM	`show module csm` *slot* `real [sfarm` *sfarm name*`]`
Веса и счетчики соединений брандмауэра	IOS	`show ip slb reals detail`
	CSM	`show module csm` *slot* `real [sfarm` *sfarm-name*`] detail`
Состояние группы брандмауэров	IOS	`show ip slb firewallfarm`

Функция отображения	Операционная система коммутатора	Команда

	CSM	`show module csm slot serverfarms [name serverfarm_name] [detail]`
		или
		`show module csm slot vservers [detail]`

SLB-соединения с брандмауэрами	IOS	`show ip slb conns [firewall firewall] [vrf-name] [detail]`
	CSM	`show module csm slot conns [vserver virtserver-name] [client ip-address] [detail]`
Тесты	IOS	`show ip slb probe [name probe_name] [detail]`
	CSM	`show module csm slot probe [http | icmp | telnet | tcp | ftp | smtp | dns] [name probe_name] [detail]`
Sticky соединения	IOS	`show ip slb sticky`
	CSM	`show module csm slot sticky [groups client ip_address]`

10.3: SLB-тесты

- Тесты (probes) могут использоваться для тестирования связи и соответствующего режима работы сервера или брандмауэра.
- Тесты могут быть определены для имитации запросов нескольких протоколов:
  - **протокол ICMP** — отправка ICMP-эхо-запросов (ping) реальному серверу;
  - **протокол HTTP** — отправка HTTP-запросов реальному серверу;
  - **протокол WSP** — инициирование запросов и проверка ответов с помощью *протокола беспроводных приложений (Wireless Application Protocol — WAP)*, порт 9201;
  - **протокол Telnet** — открытие и закрытие Telnet-соединения (TCP порт 23) с реальным сервером;
  - **протокол TCP** — установка и переустановка TCP-соединений с реальным сервером. этот тест может использоваться для поддержки любого TCP-порта, включая HTTPS и SSL, порт 443;
  - **протокол FTP** — открытие и закрытие FTP-соединения (TCP-порт 20 и 21) с реальным сервером;
  - **протокол SMTP** — открытие и закрытие SMTP-соединения (TCP-порт 25) с реальным сервером;
  - **протокол DNS** — отправка запросов к реальному DNS-серверу и проверка поступающих от него ответов.

Конфигурирование службы

1. Указание теста.

IOS-команда	`ip slb probe` *name* `{ping | http | wap}` (режим глобальной конфигурации)
CSM-команда	`probe` *probe name* `{http | icmp | telnet | tcp | ftp | smtp | dns}` (режим конфигурирования модуля CSM)

Тесту присваивается имя (параметр *name* — текстовая строка длиной до пятнадцати символов), по которому на него можно сослаться из других команд серверной группы или группы брандмауэров SLB. В службе SLB операционной системы IOS допускаются следующие типы тестов: `ping` (ICMP), `http` и `wap` (WAP, порт 9201). В CSM модуле в дополнение к указанным допускаются тесты `http`, `icmp` (ping), `telnet`, `tcp` (TCP-соединение), `ftp`, `smtp` и `dns`.

2. Определение целевого адреса (*необязательно*)

IOS-команда	`address` *{ip-address}* (режим конфигурирования теста)
CSM-команда	`address` *ip-address* (режим конфигурирования теста)

Для серверной группы эта команда не используется. IP адрес (параметр *ip address*), используемый в тесте, предоставляется каждым реальным сервером в серверной группе. Однако модуль CSM позволяет конфигурировать адрес для ping- или DNS-теста.

При использовании службы SLB операционной системы IOS в ситуации, когда тестируется группа брандмауэров, адреса не наследуются. Чтобы указать адрес целевого брандмауэра, необходимо использовать приведенную выше команду.

3. Установка режима теста

а) Настройка интервала между тестами

IOS-команда	`interval` *seconds* (режим конфигурирования теста)
CSM-команда	`interval` *seconds* (режим конфигурирования теста)

В направлении цели тесты отправляются с заданным интервалом (параметр *seconds*, в службе SLB операционной системы IOS 1–65 535 секунд, стандартно — 1 секунда; в модуле CSM 5–65 535 секунд, стандартно — 120 секунд).

б) Установка времени ожидания для ответа на TCP-тест (*необязательно, только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`receive` *receive-timeout* (режим конфигурирования теста)

Модуль CSM в течение определенного времени (параметр *receive-timeout* — от 1 до 65 535 секунд; стандартно — 10 секунд) ожидает получения данных в ответ на отправленную пробу, тип которой отличен от TCP.

в) Установка времени ожидания для соединения TCP-теста (*необязательно; только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`open` *open-timeout* (режим конфигурирования теста)

Для HTTP-, TCP-, FTP-, Telnet- и SMTP-тестов модуль CSM в течение определенного времени (*open-timeout* — от 1 до 65 535 секунд, стандартно — 60 секунд) ожидает открытия TCP-соединения и ответ на отправленный запрос.

г) Определение критерия сбоя (*необязательно*).

IOS-команда	`faildetect` *retry-count* (режим конфигурирования теста)
CSM-команда	`retries` *retry-count* (режим конфигурирования теста)

В службе SLB операционной системы IOS сервер брандмауэра считается вышедшим из строя, если не поступили ответы на определенное количество последовательных `ping`-запросов (параметр *retry-count* — 1–255; стандартно — 18). Модуль CSM определяет цель как выпадающую из строя, если без ответа остались запросы любого типа в количестве, заданном параметром *retry-count* (0–65 535; стандартно — 3).

д) Время ожидания перед отправкой отказавшему серверу другого теста (*необязательно; только для модуля CSM*).

IOS-команда	Нет
CSM-команда	`failed` *failed-interval* (режим конфигурирования теста)

Определив, что какой-либо сервер вышел из строя, модуль CSM прежде чем отправить другой запрос, ожидает в течение *failed-interval* секунд (от 3 до 65 535 секунд, стандартно — 300 секунд).

4. Указание операций HTTP-теста (*необязательно; только для HTTP-теста*).

а) Установка номера порта (*необязательно; только в службе SLB операционной системы IOS*).

IOS-команда	`port` *port-number* (режим конфигурирования теста)
CSM-команда	Нет

Обычно в HTTP-запросе используется порт (*port-number*) номер 80. Однако если в команде порт не указан, номер порта может быть получен от виртуального сервера. Для тестирования брандмауэра необходимо задать номер порта (параметр

port-number — 1–65 535). Чтобы тест выполнил свою функцию, целевое устройство должно иметь возможность ответить на HTTP-запрос.

б) Указание метода HTTP-теста *(необязательно)*.

IOS-команда	`request [method {get | post | head name}] [url path]` (режим конфигурирования теста)
CSM команда	`request [method {get | head}] [url path]` (режим конфигурирования теста)

Этот тест запрашивает информацию у сервера с помощью метода `get` (стандартный метод), `post`, `head` (запрашивается тип данных заголовка) или `name` (запрашиваются именованные (`name`) данные). Также может быть задан URL-адрес, указывающий серверный путь (`path` – текстовая строка URL, стандартно `"/"`).

в) Определение информации заголовка теста *(необязательно)*

IOS-команда	`header field-name [field-value]` (режим конфигурирования теста)
CSM команда	`header field-name [field-value]` (режим конфигурирования теста)

Имя заголовка теста устанавливается в поле *field name* (текстовая строка длиной до пятнадцати символов) со значением *field-value*. Между именем и значением автоматически вставляется двоеточие. Стандартно в запросе содержатся следующие заголовки:

```
Accept: */*
Connection: close
User-Agent: cisco-slb-probe/1.0
Host: virtual-IP-address
```

г) Определение параметров HTTP-аутентификации *(необязательно)*.

IOS-команда	`credentials username [password]` (режим конфигурирования теста)
CSM-команда	`credentials username [password]` (режим конфигурирования теста)

В ситуации, когда требуется HTTP-аутентификация, для теста могут быть заданы имя пользователя (параметр *username*, текстовая строка длиной до пятнадцати символов) и пароль (параметр *password*, текстовая строка длиной до пятнадцати символов).

д) Ожидание получения определенного кода состояния *(необязательно)*.

IOS-команда	`expect [status status code] [regex regular expression]` (режим конфигурирования теста)
CSM-команда	`expect status min-number [max-number]` (режим конфигурирования теста)

Реальный сервер или брандмауэр считается вышедшим из строя, если он не отвечает на HTTP-пробу или возвращает код состояния (*status-code* = 100–599; стандартно — 200), отличный от указанного. Для брандмауэров *status-code* значение следует устанавливать равным 401. Для модуля CSM код состояния должен находиться в диапазоне, заданном с помощью значений *min-number* (стандартно — 0) и *max-number* (необязательно, стандартно — 999).

В службе SLB операционной системы IOS наряду с кодом состояния также можно ожидать получения регулярного выражения. Для этого используется ключевое слово *regex* и указывается регулярное выражение (параметр *regular-expression* — текстовая строка, стандартного значения не существует). Поиск совпадения осуществляется только в первых 2920 байтах ответа на отправленную пробу.

5. Указание целевого URL-адреса (*необязательно, только для HTTP-тестов*)

IOS-команда	`url` *path* (режим конфигурирования теста)
CSM-команда	Нет

Также может быть задан URL-адрес, указывающий серверный путь (параметр *path* — текстовая строка, стандартно — "/").

6. Указание имени домена (*необязательно; только для DNS-тестов*).

IOS-команда	Нет
CSM-команда	`name` *domain name* (режим конфигурирования теста)

Для выполнения DNS-теста требуется указание доменного имени (*domain-name* — текстовая строка), которое может быть преобразовано с помощью целевого DNS-сервера.

Отображение информации о SLB-тестах

В табл. 10.3 перечислены некоторые команды коммутатора, которые можно использовать для получения полезных сведений о SLB-тестах.

Таблица 10.3. Команды для отображения сведений о SLB-тестах

Функция отображения	Операционная система коммутатора	Команда
Тесты	IOS	`show ip slb probe [name probe_name] [detail]`
	CSM	`show module csm slot probe [http | icmp | telnet | tcp | udp | ssl | dns? [name probe_name] [detail]`

Дополнительная литература

Рекомендуемые ниже источники предоставляют дополнительную информацию по темам, рассматриваемым в этой главе.

Балансировка нагрузки на серверы средствами Cisco IOS и семейства коммутаторов Catalyst 6000 (Cisco IOS Server Load Balancing and the Catalyst 6000 Family of Switches): www.cisco.com/warp/customer/cc/pd/si/casi/ca6000/tech/iosslb_wp.htm.

Протокол динамической обратной связи Cisco (The Cisco Dynamic Feedback Protocol): www.cisco.com/warp/public/cc/pd/ibsw/mslb/tech/dfp_wp.htm.

В этой главе...

- 11.1: **подавление широковещания.** В разделе описан метод предотвращения пересылки коммутатором излишних широковещательных пакетов, полученных по какому-либо порту.
- 11.2: **фильтрация протоколов.** В этом разделе рассматриваются настройки порта, предотвращающие пересылку через какие-либо порт лавинных пакетов определенного протокола.
- 11.3: **функция обеспечения безопасности портов.** В этом разделе приводится информация по настройке порта, который позволяет использовать порт только клиентам, входящим в определенный список по основанию MAC-адресов.
- 11.4: **списки доступа VLAN-сетей.** В разделе описано управление трафиком, проходящим через коммутатор второго уровня с помощью списков контроля доступа, применяемых ко VLAN-сетям.
- 11.5: **аутентификация на коммутаторе.** В разделе поясняется методика конфигурирования коммутатора для использования технологий RADIUS, TACACS и TACACS+ для аутентификации в коммутаторе.
- 11.6: **список разрешения доступа.** В разделе показано, как создать список узлов, которым разрешен доступ к коммутатору для различных административных задач (Telnet, SNMP и HTTP).
- 11.7: **конфигурация служб SSH и Telnet.** В этом разделе приводятся сведения по конфигурированию коммутатора с целью разрешения Secure Shell Telnet-регистрации.
- 11.8: **аутентификация по протоколу 802.1X.** Прежде чем предоставить доступ к сети, порт в целях аутентификации пользователя может запросить входные параметры или сертификат. В разделе описаны действия, необходимые для создания такой конфигурации порта.

Глава 11

Управление трафиком и доступом к коммутатору

Внимание!

Многие функции управления трафиком, описанные в настоящей главе, сильно зависят от аппаратного обеспечения и продуктов. При изучении предоставленного в книге материала следует обращать внимание на то, что многие команды отличаются в разных линейках продуктов, а некоторые описанные функции не поддерживаются

11.1: подавление широковещания

- Известно, что сетевой протокол способен создавать большой объем широковещательного трафика
- В сетях второго уровня широковещательные фреймы должны перенаправляться по всем портам, кроме порта, принявшего их. Вследствие этого множество широковещательных фреймов может влиять на производительность сети и устройства.
- Подавление широковещания позволяет управлять обработкой чрезмерного широковещательного трафика на принимающем порту.
- Путем конфигурирования порогового значения какой-либо порт можно настроить на остановку лавинного распространения широковещательных фреймов на определенный период времени или до тех пор, пока уровень широковещательных рассылок не понизится до определенного значения.
- Подавление широковещательных фреймов предотвратит их передачу через остальные порты коммутатора, а также ограничит их влияние на сеть.
- Подавление широковещания не оказывает никакого влияния на многоадресатный и одноадресатный трафики, получаемые на данном порту
- Функция подавления широковещания поддерживается обеими операционными системами — как COS, так и IOS — на большинстве платформ Catalyst
- На некоторых платформах в дополнение к подавлению широковещания можно также настроить функцию подавления многоадресатного и одноадресного трафика.

Конфигурирование функции подавления широковещания

Стандартно на всех платформах и во всех операционных системах подавление широковещания отключено. Подавление широковещания применяется к отдельным портам коммутаторов. При конфигурировании этой функции следует учитывать количество широковещательных фреймов, принимаемых портом. По достижении порогового значения порт останавливает пересылку широковещательных пакетов объединительной плате до тех пор, пока заданное условие является корректным. Для настройки функции подавления широковещания используется описанная ниже последовательность действий.

1. Включение функции подавления широковещания.

Система COS	`set port broadcast` *mod/port threshold%*
IOS коммутатора 3900XL	`port storm-control broadcast threshold rising` *pprisingthreshold%* `falling` *ppfallingthreshold%* (режим конфигурирования интерфейса)
IOS коммутатора 2950	`storm-control broadcast level` *risingthreshold%* `[`*falling threshold%*`]` (режим конфигурирования интерфейса)
IOS коммутатора 3550	`storm-control broadcast level` *threshold%* (режим конфигурирования интерфейса)
Supervisor IOS (Catalyst 6000)	`broadcast suppression` *threshold%* (режим конфигурирования интерфейса)

Синтаксис для настройки этой функции и действия на разных платформах имеют отличия. Для COS-коммутаторов установка на интерфейсе порогового значения (*threshold%*) на уровне менее ста процентов включает подавление широковещания. Пороговое значение ограничивает величину полосы пропускания интерфейса, которая может использоваться для широковещательных пакетов. Например, уровень 50 процентов означает, что на данном интерфейсе нужно воздействовать на все широковещательные пакеты, превышающие 50 процентов общей полосы пропускания. Уровень 100 процентов отключает подавление широковещания, поскольку позволяет широковещательному трафику занимать до ста процентов доступной полосы пропускания. Для коммутаторов моделей 2950, 3550 и Supervisor IOS на коммутаторе 6000-й серии используется та же концепция. Параметры *threshold%* и *risingthreshold%* определяют процент ограничения полосы пропускания, по достижении которого к широковещательному трафику будет предпринято действие (action).

Для коммутаторов 2900/3500XL предел нарастания (rising value), подавляющий широковещание, определяется количеством *пакетов в секунду* (pps). По достижении уровня *pprisingthreshold#* по отношению к пакетам применяется действие. Для указанных коммутаторов необходимо также настроить значение *ppfallingthreshold#*, определяющее уровень, ниже которого должен опуститься широковещательный трафик, чтобы с данного порта было снято ограничение.

Следует помнить о том, что 100 Мбит/с Ethernet-порт обладает максимальной пропускной способностью 149 000 pps в дуплексном режиме. Значение, равное или превышающее 149 000 pps, на управление широковещанием не влияет.

В COS-коммутаторах, во всех операционных системах на коммутаторах серии 6000 и коммутаторах модели 3550, подавление широковещания является периодическим действием, т.е. широковещательные пакеты подавляются с интервалом в одну секунду. В коммутаторах 2950 и 2900XL подавление основано на абсолютном значении или пределе падения (falling value). `fallingthreshold` — необязательное значение, которое может быть установлено для того, чтобы указать, когда с порта снимается ограничение. Например, если порт `risingthreshold` равен пятидесяти процентам, а `fallingthreshold` — сорока пяти, то действие применяется, когда 50 процентов трафика составляют широковещательные пакеты. Это действие применяется до тех пор, пока уровень широковещания не опустится ниже сорока пяти процентов. Поскольку значение предела падения не является обязательным, то, если оно не задано, по умолчанию ограничение снимается, когда процент широковещательного трафика опустится ниже предела нарастания.

Внимание!

Подавление широковещания не поддерживается на коммутаторах Catalyst серии 4000.

2. Указание предпринимаемого действия.

Система COS	`set port broadcast` *mod/port threshold%* `[violation {drop-packets | errdisable}]`
IOS коммутатора 3500XL	`port storm-control broadcast action {filter | shutdown}` (режим конфигурирования интерфейса)
IOS коммутатора 2950	`storm-control broadcast action {shutdown | trap}` (режим конфигурирования интерфейса)
IOS коммутатора 3500	Нет
Supervisor IOS (Catalyst 6000)	Нет

В ситуации, когда широковещание подавляется, стандартными действиями являются удаление или фильтрация пакетов. В данном случае это означает, что пакеты удаляются и не доходят до объединительной платы коммутатора. На некоторых платформах можно настроить другое действие. Например, в COS-коммутаторах можно перевести порт в состояние *errdisable*. Это означает, что порт будет оставаться в таком состоянии всё время, пока уровень будет равен пороговому значению. В таком случае порт будет находиться в состоянии *errdisable* даже после того, как уровень широковещания понизится, до тех пор, пока администратор не устранит проблему. В коммутаторе серии 3500XL можно стандартное действие устройства `filter` изменить на `shutdown`. После перевода в режим `shutdown` порт остается отключенным до тех пор, пока администратор не включит его. Каждый раз при превышении порогового значения администратору необходимо включать порт. В коммутаторах модели 2950, если действие не

отключено на `shutdown`, фреймы отбрасываются; эта функция работает так же, как и в коммутаторах серии 3500XL. Чтобы вернуться к фильтрации фреймов, администратору необходимо ввести команду `no port storm-control broadcast action shutdown`. Другой вариант конфигурации коммутатора 2950 заключается в том, что коммутатор генерирует SNMP прерывание (`trap`). Это действие невозможно настроить на коммутаторах серии 2550 и Catalyst 6000, использующих Supervisor IOS.

Внимание!

С помощью команды `set errdisable-timeout enable bcast-suppression` коммутатор можно настроить на автоматическое включение порта, переведенного вследствие широковещательной лавины в состояние errdisable

3. Управление одноадресатными и многоадресатными рассылками (*необязательно*).

Система COS	`set port broadcast` *mod/port. threshold%* `[multicast {enable | disable}] [unicast {enable | disable}]`
IOS коммутатора 3500XL	`port storm-control {multicast | unicast} threshold rising` *pps risingthreshold# falling pps fallingthreshold#* (режим конфигурирования интерфейса)
IOS коммутатора 2950	`storm-control {multicast unicast} level` *risingthreshold% [fallingthreshold%]* (режим конфигурирования интерфейса)
IOS коммутатора 3500	`storm-control {multicast unicast} level` *threshold%* (режим конфигурирования интерфейса)
Supervisor IOS (Catalyst 6000)	Нет

В дополнение к конфигурированию коммутатора, для управления широковещательными лавинами можно настроить какой-либо порт на отбрасывание фреймов или отключение при возникновении большого количества одноадресатных или многоадресатных пакетов. Чтобы настроить эту функцию, в командах используются ключевые слова `multicast` и `unicast`, активизирующие управление фреймами.

Проверка конфигурации

После того как подавление широковещаний настроено для проверки конфигурации и правильности функционирования коммутатора, используются приведенные ниже команды

Система COS	`show port broadcast` *[mod] [/port]*
IOS коммутатора 3500XL	`show port storm-control` *{interface}* (режим привилегированного пользователя)

IOS коммутатора 2950/3500	`show storm-control [interface] [{broadcast | multicast | unicast | history}]` (режим привилегированного пользователя)
Supervisor IOS (Catalyst 6000)	`show interfaces switchport [module number]` (режим привилегированного пользователя)

Пример конфигурирования функции

В этом примере демонстрируется типичная конфигурация подавления широковещания на порту 3/1 COS-коммутатора с пороговым значением, равным тридцати трем процентам, по достижении которого порт переводится в состояние errDisable. В то же время задана такая конфигурация, при которой коммутатор пытается автоматически включить данный порт.

Ниже приведен пример конфигурации Catalyst OS.

```
Catalyst (enable)> set port broadcast 3/1 33% violation errdisable
Catalyst (enable)> set errdisable-timeout enable bcast-suppression
```

Для коммутатора 2950 в данном примере демонстрируется конфигурация, активизирующая подавление широковещания в случае, если трафик займет 55 процентов полосы пропускания интерфейса Fast Ethernet 0/9, и восстанавливающая обычное распространение широковещательных фреймов после падения объема такого трафика ниже сорока четырех процентов.

Ниже приводится пример конфигурации Supervisor IOS.

```
2950(config)#interface fastethernet 0/9
2950(config-if)#storm-control broadcast level 55 44
2950(config-if)#end
2950#copy running-config startup-config
```

11.2: фильтрация протоколов

- Фильтрация протоколов может быть настроена на коммутаторах Catalyst серий 4000, 5000 и 6000.
- Для реализации этой функции не требуются на коммутаторе какие-либо специальные функциональные платы.
- Фильтрация протоколов позволяет конфигурировать порт для фильтрации или блокировки данного трафика (широковещательного, многоадресатного и одноадресатного трафика с неизвестным получателем) на основании протоколов.
- Фильтрация протоколов поддерживается только на портах доступа второго уровня. Эту функцию невозможно настроить на магистральных каналах и портах третьего уровня.
- Фильтрация протоколов поддерживает блокировку потоков данных IP, IPX, AppleTalk, VINES и DECnet. На трафик всех остальных протоколов эта функция не влияет.
- Административные протоколы, такие, как *протокол распределенного остовного дерева (Spanning Tree Protocol — STP)*, *протокол обнаружения устройств Cisco (Cisco Discovery Protocol — CDP)* и *протокол магистральных каналов VLAN-сетей (VLAN Trunking Protocol — VTP)*, этой функцией не блокируются

Конфигурирование функции

Фильтрация протоколов на коммутаторе не позволяет порту лавинно распространять через заданный порт трафик данного типа, полученный от других портов VLAN-сети. Эта функция может быть полезной при управлении трафиком внутри одной VLAN-сети от клиентов, которые используют различные протоколы, в том числе протоколы с интенсивным обменом данными. Ниже описаны этапы конфигурирования фильтрации протоколов.

1. Включение фильтрации протоколов на коммутаторе.

Система COS	`set protocolfilter enable`
Система IOS	`protocol-filter` (режим глобальной конфигурации)

Фильтрация протоколов стандартно отключена. Чтобы обеспечить управление трафиком на определенных портах, в первую очередь фильтрацию протоколов необходимо включить на коммутаторе. После этого можно настроить реакцию портов на заданный протокол.

2. Включение фильтрации протоколов на порту доступа к сети

Система COS	`set port protocol mod/port {ip | ipx group} {on | off | auto}`
Система IOS	`switchport protocol {ip | ipx | group} {on | off | auto}` (режим конфигурирования интерфейса)

Для каждого порта, на котором требуется обеспечить управление трафиком, необходимо указать протокол и обработку трафика. Параметр `protocol` определяет заданный тип протокола, при этом может использоваться одно из следующих ключевых слов: `ip` (IP), `ipx` (IPX), `group` (протоколы AppleTalk, DECnet и Banyan VINES). Обработка трафика определяется параметрами. Параметр `on` указывает на то, что порт используется для получения трафика данного протокола и пересылки лавинного трафика для него. Параметр `off` указывает на то, что порт не может принимать или лавинно распространять трафик для заданного протокола. Параметр `auto` указывает на то, что порт не будет лавинно распространять трафик данного протокола до тех пор, пока не получит пакет этого протокола. В табл. 11.1 приведены стандартные действия в случае, если порты не сконфигурированы.

Таблица 11.1. Стандартные настройки фильтрации протоколов

Протокол	Режим
IP	on
IPX	auto
Group	auto

Проверка конфигурации

Для проверки конфигурации фильтрации протоколов используются приведенные ниже команды.

Система COS	`show port protocol` *mod/port (s)*

Система IOS	`show protocol-filtering`
	или
	`show protocol-filtering interface {type slot/port}`
	(обе команды вводятся в режиме привилегированного пользователя)

Эти команды **show** позволяют отобразить конфигурацию определенных портов. В операционной системе IOS команда **show protocol-filtering** без указания назначения портов отображает только порты, обрабатывающие по крайней мере один протокол в нестандартном режиме.

Пример конфигурирования функции

В этом примере демонстрируется конфигурация фильтрации протоколов. В начале примера используется команда включения фильтрации протоколов. Затем порты Fast Ethernet с 5/1 по 5/6 настраиваются на прохождение IP-трафика без фильтрации, а также на блокировку всех остальных протоколов. Также порты 5/7 и 5/8 настраиваются на передачу только IPX-трафика. В этом примере на портах 5/9 и 5/10 разрешена передача IP- и IPX-трафика только в случае обнаружения на этих портах IP- или IPX-клиента, а также разрешена пересылка всего остального трафика.

Ниже приведен пример конфигурации Catalyst OS.

```
Catalyst(enable)>set protocolfilter enable
Catalyst(enable)>set port protocol 5/1-6 ip on
Catalyst(enable)>set port protocol 5/1-6 ipx off
Catalyst(enable)>set port protocol 5/1-6 group off
Catalyst(enable)>set port protocol 5/7-8 ip off
Catalyst(enable)>set port protocol 5/7-8 ipx on
Catalyst(enable)>set port protocol 5/7-8 group off
Catalyst(enable)>set port protocol 5/9-10 ip auto
Catalyst(enable)>set port protocol 5/9-10 ipx auto
Catalyst(enable)>set port protocol 5/9-10 group on
```

Пример конфигурации Supervisor IOS

```
Switch(config)#protocol-filter
Switch(config)#interface fastethernet 5/1
Switch(config-if)#switchport protocol ip on
Switch(config-if)#switchport protocol ipx off
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/2
Switch(config-if)#switchport protocol ip on
Switch(config-if)#switchport protocol ipx off
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/3
Switch(config-if)#switchport protocol ip on
Switch(config-if)#switchport protocol ipx off
Switch(config-if)#switchport protocol group off
```

```
Switch(config-if)#interface fastethernet 5/4
Switch(config-if)#switchport protocol ip on
Switch(config-if)#switchport protocol ipx off
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/5
Switch(config-if)#switchport protocol ip on
Switch(config-if)#switchport protocol ipx off
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/6
Switch(config-if)#switchport protocol ip on
Switch(config-if)#switchport protocol ipx off
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/7
Switch(config-if)#switchport protocol ip off
Switch(config-if)#switchport protocol ipx on
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/8
Switch(config-if)#switchport protocol ip off
Switch(config-if)#switchport protocol ipx on
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/9
Switch(config-if)#switchport protocol ip auto
Switch(config-if)#switchport protocol ipx auto
Switch(config-if)#switchport protocol group off
Switch(config-if)#interface fastethernet 5/10
Switch(config-if)#switchport protocol ip auto
Switch(config-if)#switchport protocol ipx auto
Switch(config-if)#switchport protocol group off
Switch(config-if)#end
Switch(config)#copy running-config startup-config
```

11.3: функция обеспечения безопасности портов

- Функция обеспечения безопасности портов позволяет настроить какой-либо порт так, чтобы доступ к коммутатору через этот порт предоставлялся только заданному устройству или устройствам.
- В функции обеспечения безопасности портов устройства, которым разрешается доступ, определяются с помощью MAC-адресов.
- MAC-адреса для разрешенных устройств могут быть настроены вручную и/или "изучены" коммутатором.
- Существуют ограничения на количество MAC-адресов, которые обслуживаются портом в режиме безопасности. Такие ограничения различны для разных платформ.
- При обнаружении попытки доступа к порту с неавторизованного MAC-адреса коммутатор может приостановить работу порта или отключить его.
- Эту функцию невозможно настроить на магистральном порту, SPAN-порту (*Switched Port Analyzer — анализатор коммутируемых портов*) или на порту, динамически назначаемом какой-либо VLAN-сети.

- Безопасность портов поддерживается на коммутаторах серии 5000, 4000 и 6000, использующих операционную систему COS, а также на коммутаторах 3500XL, 3550 и 2950 с системой IOS.

Конфигурирование функции

Если какой-либо порт коммутатора активен, любой пользователь может подключиться к нему и получить доступ к сети. Поскольку во многих сетях для выделения пользовательских адресов используется *протокол динамического конфигурирования узла (Dynamic Host Configuration Protocol — DHCP)*, для любого лица, имеющего физический доступ к сетевому порту, могло бы быть весьма просто подключить свое устройство, такое, как портативный компьютер, к этому порту и стать пользователем сети. С этого порта такой пользователь мог бы продолжать генерировать трафик или создавать другие проблемы внутри сети. Функция обеспечения безопасности портов позволяет указать MAC-адрес (или адреса) устройств, которым разрешено подключаться к порту. Этапы конфигурирования функции описаны ниже.

1. Включение функции обеспечения безопасности портов.

Система COS	`set port security` `mod/port enable`
Система IOS	`switchport port-security` (режим конфигурирования интерфейса)
IOS коммутатора 3500XL	`port security` (режим конфигурирования интерфейса)

Стандартно любой пользователь может подключиться к порту и получить доступ к сетевым службам. Для защиты порта прежде всего необходимо включить функцию безопасности. При этом рекомендуется применять команду, соответствующую используемому устройству.

Внимание!

На момент написания этой книги возможность настроить безопасность портов на коммутаторах серий 4000 и 6000, использующих Supervisor IOS, отсутствовала. Ожидается, что если корпорация Cisco обеспечит поддержку данных платформ, то синтаксис будет подобен конфигурации IOS (а не системе IOS устройств 3500XL), приведенной при описании этих этапов.

2. Указание количества MAC-адресов.

Система COS	`set port security` `mod/port maximum value`
Система IOS	`switchport port-security maximum value` (режим конфигурирования интерфейса)
IOS коммутатора 3500XL	`port security max-mac-count value` (режим конфигурирования интерфейса)

После включения функции безопасности портов необходимо определить количество различных устройств, которые получат доступ к этим портам, а также количество адресов, нуждающихся в обеспечении безопасности. Количество этих адресов определяется параметром `value`, стандартное значение которого равно

единице. Каждая аппаратная платформа имеет ограниченное число адресов, для которых можно обеспечить безопасность портов, поэтому если на коммутаторе требуется обеспечить безопасность более чем двухсот пятидесяти адресов, то следует ознакомиться со специальной документацией на используемое аппаратное обеспечение.

3. Ввод вручную MAC-адресов, для которых необходимо обеспечить безопасность.

Система COS	`set port security` *mod/port* `enable` [*mac_address*]
Система IOS	`switchport port-security mac-address` *mac_address* (режим конфигурирования интерфейса)
IOS коммутатора 3500XL	Нет

Стандартно коммутаторы "изучают" MAC-адреса устройств, подключенных к данному порту. Если требуется обеспечить контроль над устройствами, которые могут получать доступ к коммутатору, то для указания MAC-адресов, безопасных для порта, следует использовать описанные выше команды.

4. Определение действия, предпринимаемого портом.

Система COS	`set port security` *mod/port* `violation {shutdown | restrict}`
Система IOS	`switchport port-security violation {protect | restrict | shutdown}` (режим конфигурирования интерфейса)
IOS коммутатора 3500XL	`port security action {shutdown | trap}` (режим конфигурирования интерфейса)

При нарушении указанного в конфигурации правила коммутатор обычно защищает порт, отбрасывая трафик, поступающий с незарегистрированных MAC-адресов. Это означает, что коммутатор не допускает прохождение через устройство таких фреймов, но в то же время допускается прохождение фреймов, поступающих от устройства, которое сконфигурировано как безопасное. Такие установки являются стандартными для каждого устройства и указываются с помощью параметра `protect` в IOS-коммутаторах и `restrict` – в COS-коммутаторах. Параметр `restrict` является стандартным при включении безопасности портов на коммутаторе 3500XL (если не указан какой-либо иной параметр). Другая возможность при конфигурировании заключается в переводе интерфейса в состояние `shutdown`. Если эта возможность используется, порт остается в состоянии административного отключения до тех пор, пока администратор не включит его с помощью команды `no shutdown`. Третьим вариантом является генерирование SNMP-прерываний. Эту функцию при нарушении безопасности осуществляют параметры `restrict` — в IOS-устройствах и `trap` в IOS-коммутаторах 3500XL.

Проверка конфигурации

Для проверки конфигурации функции обеспечения безопасности портов используются приведенные ниже команды.

Система COS	`show port security [statistics] mod/port` `show port security statistics [system] (mod/port)`
Система IOS	`show port security [interface interface-id] [address]` (режим привилегированного пользователя)
IOS коммутаторы 3500XL	`show port security [interface-id]`

Пример конфигурирования функции

В этом примере демонстрируется конфигурация функции обеспечения безопасности портов. Порт Fast Ethernet 2/1 конфигурируется так, чтобы доступ к нему был разрешен только с единственного MAC-адреса — 00-01-03-87-09-43. Кроме того, настраивается отключение порта в случае нарушения безопасности. Порты 2/2 и 2/3 настраиваются таким образом, что к каждому из них разрешен доступ десяти адресов, которые будут определены коммутатором при подключении устройств к данным портам. Неавторизованные пакеты отбрасываются.

Ниже приведен пример конфигурации Catalyst OS.

```
Catalyst (enable)>set port security 2/1 enable
Catalyst (enable)>set port security 2/1 enable 00-01-03-87-09-43
Catalyst (enable)>set port security 2/1 violation shutdown
Catalyst (enable)>set port security 2/2-3 enable
Catalyst (enable)>set port security 2/2-3 maximum 10
```

Ниже приводится пример конфигурации для Supervisor IOS.

```
Switch(config)#interface fastethernet 2/1
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security mac-address 00-01-03-87-09-43
Switch(config-if)#switchport port-security violation shutdown
Switch(config)#interface fastethernet 2/2
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security maximum 10
Switch(config)#interface fastethernet 2/3
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security maximum 10
Switch(config-if)#end
Switch(config)#copy running-config startup-config
```

11.4: списки доступа VLAN-сетей

- *Списки контроля доступа (Access Control List — ACL)* определяют то, каким образом трафик обрабатывается при прохождении через сетевое устройство.
- В списках ACL для управления обменом данными используется адресная информация и сведения о портах.
- Как правило, списки ACL реализуются в маршрутизаторах, однако новое аппаратное обеспечение позволяет коммутаторам второго и третьего уровней перед отправкой пакетов обращаться к этим спискам.

- Списки ACL позволяют пользователям настраивать любой коммутатор для управления трафиком на основании третьего или вышестоящих уровней эталонной модели OSI.
- Списки ACL для управления потоками данных связаны с VLAN-сетью или портом второго уровня.
- Списки контроля доступа VLAN-сетей (VACL) управляются аппаратным обеспечением и поддерживаются не всеми платформами.
- В настоящее время VACL поддерживаются на коммутаторах серий 6000 с *функциональной платой политики - Policy Feature Card* (PFC или PFC2), 3550 и 2950.

Списки VLAN ACL, VACL — список контроля доступа, который определяет параметры трафика на основании информации третьего и более высоких уровней и применяется ко VLAN-сети второго уровня или в некоторых случаях к интерфейсу второго уровня. Такие списки, реализуемые аппаратным обеспечением, обладают определенными преимуществами перед традиционными списками доступа маршрутизаторов и, следовательно, являются более скоростными, чем традиционные ACL. Кроме того, они предоставляют возможность фильтрации трафика внутри IP-подсети и за ее пределами. Несмотря на то что функциональность ACL одинакова в различных операционных системах, их конфигурация отличается. Этот раздел разделен на две части. Первый блок команд определяет конфигурацию VACL на COS-устройствах, поддерживающих списки VACL. Во второй части определяются списки VACL в системе IOS. Чтобы сконфигурировать VACL-списки и применить их на конкретном коммутаторе, следует использовать этапы в обеих частях. Описанные ниже действия применяются только к IP VACL-спискам, поскольку данный протокол поддерживается на всех перечисленных платформах. На некоторых платформах существует возможность настроить списки VACL для протокола IPX. Хотя синтаксис и процессы являются одинаковыми, параметры протокола для IPX отличаются от описанных.

Внимание!

Режим работы списков ACL одинаков как на маршрутизаторах, так и на коммутаторах. В этом разделе не обсуждаются все параметры и ключевые моменты конфигурации. Более подробная информация по конфигурации списков доступа приведена в изданиях Cisco Press *Сетевое взаимодействие устройств Cisco (Interconnecting Cisco Network Devices)* и *Практическое руководство Cisco. конфигурирование маршрутизаторов (Cisco Field Manual. Router Configuration)*.

Конфигурирование списков VACL в системах COS

Описанные ниже действия применяются к коммутатору Catalyst серии 6000 с платой PFC или PFC2, использующей программное обеспечение операционной системы COS.

1. Конфигурирование списка доступа

Система COS	`set security acl ip {acl_name} {permit deny | redirect mod/port} {protocol} {source address mask} [op] [sport] [dest mask] [op] [dport] [before editbuffer index | modify editbuffer index] [log]`

Для управления потоками данных прежде всего необходимо определить трафик, который требуется контролировать, и его обработку. Список *VACL* — это последовательный список записей, которые определяют контролируемый трафик и способ управления им. Чтобы создать такой список, необходимо ввести приведенные выше команды для каждой группы проверяемых условий. Если фрейм соответствует критерию какой-либо записи списка (в направлении сверху вниз), к нему применяется определенное действие. Если записи, соответствующей полученному фрейму, нет, фрейм отбрасывается.

Команда `set security acl ip` *name* определяет имя конфигурируемого IP ACL. После указания имени все проверяемые записи будут иметь то же имя. После определения имени следует спецификация протокола и действий. Действие определяется первым объектом после имени. Параметр `permit` позволяет пропускать через коммутатор трафик, соответствующий заданным спецификациям. Параметр `deny` приводит к уничтожению пакетов, соответствующих спецификациям. Параметр `redirect` отправляет трафик указанному порту (*mod/port*) коммутатора вместо использования записи таблицы в памяти, адресуемой по содержимому (CAM).

Параметры `protocol`, *address/mask*, `ports` и `operator` позволяют определить поток данных по IP-адресу и сведениям, касающимся порта. Вместо параметра *address/mask* можно использовать ключевое слово `all`.

Стандартно все записи, создаваемые в ACL, вносятся в конец списка. Однако в COS-устройстве список ACL создается и специальном буфере редактирования и не является функциональной частью коммутатора до тех пор, пока не будет подтвержден с помощью ключевого слова `committed`. Можно поместить новый объект перед определенной записью (ключевое слово `before`) в буфере редактирования или модифицировать (заменить) указанную запись (`modify`). Для просмотра записей в буфере редактирования, который еще не был записан в память, используется команда `show security acl ip` *name* `editbuffer`. Затем для изменения или ввода новых записей используются индексные номера буферов редактирования, отображаемые с помощью указанной выше команды. Последний параметр, `log`, позволяет коммутатору протоколировать фреймы, которые были запрещены списком доступа.

2. Ввод ACL в *тернарную адресуемую по содержимому память (Ternary Content Addressable Memory — TCAM)*.

Система COS	`commit security acl {`*name* `| all}`

После того как список VACL сконфигурирован, он только резидентно присутствует в буфере редактирования и не может использоваться коммутатором до тех пор, пока не будет записан в TCAM таблицы. Чтобы внести эту информацию в TCAM-таблицу для последующего использования коммутатором, применяется команда `commit security acl`. Параметр `all` позволяет записать все неподтвержденные VACL-списки, а параметр *name* подтверждает только указанный список.

3. Назначение ACL-списка VLAN-сети.

Система COS	`set security acl map acl` *name vlan*

После того как VACL-список был создан и подтвержден и прежде чем с его помощью будет контролироваться трафик для коммутатора, необходимо назначить этот список VLAN-сети. Чтобы это сделать, нужно использовать команду **set security acl map** *acl_name vlan*, указав имя VACL и соответствующей VLAN-сети. Каждой VLAN-сети в целях управления трафиком может соответствовать только один список VACL, однако любой список может относиться ко множеству VLAN.

Проверка конфигурации

Для проверки конфигурации и привязки COS VACL-списков на коммутаторе используется одна из перечисленных ниже команд.

Система COS	**show security acl info** [*name*	**all**] **show security acl map** [*name*	*vlan*	**all**]

Конфигурация списков VACL системы IOS

В операционной системе IOS списки VACL конфигурируются как стандартные или расширенные списки IP-доступа. Затем эти списки назначаются порту или VLAN-сети. В настоящее время VACL-списки поддерживаются только на коммутаторах серии 6000, использующих Supervisor IOS, а также на коммутаторах серии 3550 и 2950G. Для настройки параметров VACL-списков применяются описанные ниже команды.

1. Конфигурирование списка доступа.

   Первым параметром, который необходимо сконфигурировать, является список, идентифицирующий трафик, контролируемый с помощью этого списка. Для системы IOS ACL-список может быть либо пумерованным, либо именованным. Кроме того, существуют различные типы ACL, например: стандартные списки, в которых указывается информация об отправителе, и расширенные, определяющие отправителя и получателя данных. Чтобы сконфигурировать списки доступа, рекомендуется использовать описанную ниже последовательность команд.

   а) Конфигурирование пумерованного стандартного списка доступа

Система IOS	**access-list** *access-list-number* {**deny** | **permit** | **remark**} {*source source-wildcard* | **host** *source* | **any**} (режим глобальной конфигурации)

   С помощью приведенной команды создается стандартный список доступа. Диапазон номеров для стандартных списков ACL: 1-99 и 1300-1999. Параметр **permit** разрешает прохождение трафика, а **deny** запрещает. Параметр **remark** позволяет вставлять в список комментарии, которые предоставляют информацию об этом списке, и причины добавления параметров. Параметр *address/mask* для ключевых слов **permit** и **deny** позволяет контролировать трафик с указанных адресов отправителей. Для указания всех адресов отправителей используется ключевое слово **any**.

б) Конфигурирование нумерованного расширенного списка доступа

Система IOS	**access-list** *access-list-number* {**deny** , **permit** ! **remark**} *protocol* {*source source-wildcard* | **host** *source* | **any**} {*operator port*} {*destination destination wildcard* | **host** *destination* | **any**} {*operator port*} (режим глобальной конфигурации)

С помощью этой команды создается расширенный список доступа. Номера для расширенных ACL-списков находятся в диапазоне 100-199 и 2000-2699. Параметр **permit** разрешает прохождение трафика, а **deny** запрещает. Параметр **remark** позволяет вставлять в список комментарии, предоставляющие информацию о данном списке, и причины добавления параметров

Параметр *protocol* указывает тип IP-протокола для проверки, например, **udp** или **tcp**. Ключевое слово **ip** в этом поле соответствует всему IP-трафику. Пара параметров *address/mask* определяет адреса отправителя и получателя передающего и принимающего устройств, для которых осуществляется управление трафиком. Чтобы указать все адреса отправителей или получателей, можно использовать ключевое слово **any**. Параметры *operator* и *port* позволяют задать специфические для протокола и приложения порты

в) Конфигурирование именованного стандартного списка доступа

Система IOS	**ip access-list standard** {*name*} (режим глобальной конфигурации) {**deny** | **permit**} {*source source-wildcard* | **host** *source* | **any**}

При работе со стандартным именованным ACL-списком команда **ip access-list standard** *name* сообщает системе о том, что необходимо войти в конфигурационный режим для списка, указанного с помощью заданного имени. Из этого режима коммутатор перейдет в режим, позволяющий последовательно вводить параметры до тех пор, пока пользователь не покинет конфигурационный режим ACL.

Параметр **permit** разрешает прохождение трафика, а **deny** запрещает. Пара параметров *address/mask* с ключевыми словами **permit** и **deny** позволяет контролировать трафик, поступающий с заданных адресов. Для указания всех адресов отправителей можно использовать ключевое слово **any**.

г) Конфигурирование именованного расширенного списка

Система IOS	**ip access-list extended** {*name*} (режим глобальной конфигурации) {**deny** ! **permit**} *protocol* {*source source-wildcard* | **host** *source* | **any**} {*operator port*} {*destination destination-wildcard* | **host** *destination* | **any**} {*operator port*}

При работе с расширенным именованным ACL-списком команда **ip access-list extended** *name* сообщает системе о том, что необходимо войти в режим

конфигурирования списка, указанного с помощью заданного имени. Из этого режима коммутатор перейдет в режим, позволяющий построчно вводить параметры до тех пор, пока пользователь не покинет конфигурационный режим ACL.

Параметр `permit` разрешает применение трафика, а `deny` запрещает. Параметр *protocol* указывает тип IP-протокола для проверки, например, `udp` или `tcp`. Ключевое слово `ip` в этом поле соответствует всему IP-трафику. Пара значений *address/mask* определяет адреса отправителя и получателя передающего и принимающего устройств, для которых осуществляется управление трафиком. Для указания всех адресов отправителей и получателей можно использовать ключевое слово `any`. Параметры *operator* и *port* позволяют задать специфические для протокола или приложения порты.

2. Создание VLAN-таблицы.

Если созданный список предполагается назначить VLAN-сети, то необходимо внести команду `vlan access-map`, чтобы задать имя таблицы доступа (access map) и действие, применяемое к определенной соответствующей записи.

Система IOS	`vlan access-map` *name* [*number*] (режим глобальной конфигурации) `match ip address` {*aclname*	*aclnumber*} `action` {`drop`	`forward`}

Таблица доступа (access map) представляет собой список операторов преобразования (map clauses), определяющих действие, применяемое к пакетам в данной VLAN-сети. При создании таблицы доступа ей назначается имя, а затем пумеруются последующие операторы. Каждый оператор проверяется в поисках соответствия в пакетах, а затем к ним применяется действие, указанное для данного оператора. Если для пакета не найдено ни одного оператора, он отбрасывается. Для создания таблицы доступа используется команда `vlan access-map` с указанием имени (name). Параметр *number* используется для последующих операторов в таблице доступа.

После ввода имени таблицы коммутатор переходит в конфигурационный режим таблицы доступа, где указывается имя ACL списка или номер для идентификации трафика, на который будет воздействовать оператор. Для списков доступа, включенных в таблицу, директива `permit` в ACL соответствует, а `deny` не соответствует заданному оператору преобразования. После того как соответствие будет идентифицировано ACL, с помощью команды `action` задается действие над трафиком (разрешение или отбрасывание). Если ни один из операторов не соответствует этому фрейму, он отбрасывается.

3. Применение списков доступа.

После создания списка доступа его необходимо применить в коммутаторе. Применение списков для разных платформ отличается. В коммутаторе серии 6000, использующем IOS, и коммутаторе 3550 для привязки списка к VLAN-сети используется параметр *a*. Если список доступа конфигурируется на коммутаторе 2950, то для применения списка к интерфейсу используется параметр *b*.

а) Применение VLAN-таблицы ко VLAN-сети.

Система IOS	`vlan filter mapname vlan-list list` (режим глобальной конфигурации)

Чтобы применить таблицу доступа ко VLAN-сети в IOS-коммутаторах, поддерживающих VACL, используется команда `vlan filter`. Параметр `mapname` определяет имя таблицы, созданной на этапе 2. За параметром `vlan-list` следует номер VLAN сети или список VLAN-номеров, к которым применим этот ACL-список.

б) Применение списка к интерфейсу.

Система IOS	`ip access-group {access-list-number | name} in` (режим конфигурирования интерфейса)

В коммутаторах 2950 ACL-списки применяются к интерфейсу второго уровня. В таких списках доступ прохождение трафика разрешается или запрещается с помощью записей списка и без использования операторов преобразования. Чтобы применить список к интерфейсу, используется команда `ip access-group`, за которой следует номер или ACL-имя. Для ACL-списков второго уровня параметром `in` определяется направление, в котором ACL применяется к интерфейсу. Пакеты могут проверяться только при поступлении в коммутатор.

Проверка конфигурации

Для проверки конфигурации в системе IOS VACL-списков используются приведенные ниже команды:

Система IOS	`show ip access-lists {number | name}`
	`show vlan access-map {mapname}`
	`show vlan filter {access-map name | vlan vlan-id}`
	`show ip interface type number`

Пример конфигурирования функции

В этом примере демонстрируется конфигурация для VACL-фильтрации. В списке, конфигурируемом на данном коммутаторе, требуется выполнить такие условия:

- разрешить весь IP-трафик из подсети 10.101.0.0 к узлу 10.101.1.1;
- разрешить эхо-запросы ICMP, поступающие от всех узлов;
- разрешить эхо-ответы ICMP, поступающие от всех узлов;
- запретить весь остальной ICMP-трафик;
- разрешить весь TCP-трафик;
- запретить весь UDP-трафик, не указанный выше;
- разрешить весь остальной IP-трафик.

Этот список требуется применить к коммутатору ко VLAN-сети 101.
Ниже приводится пример конфигурации для системы Catalyst OS.

```
Catalyst (enable)>set security acl ip watchlist permit ip 10.101.0.0
0.0.255.255 host 10.101.1.1
Catalyst (enable)>set security acl ip watchlist permit icmp any any echo
Catalyst (enable)>set security acl ip watchlist permit icmp any any echo
Catalyst (enable)>set security acl ip watchlist permit icmp any any echo-
reply
Catalyst (enable)>set security acl ip watchlist deny icmp any any echo
Catalyst (enable)>set security acl ip watchlist permit tcp any any
Catalyst (enable)>set security acl ip watchlist deny udp any any
Catalyst (enable)>set security acl ip watchlist permit ip any any
Catalyst (enable)>commit security acl ip watchlist
Catalyst (enable)>set security acl ip map watchlist 101
```

Ниже приводится пример конфигурации Supervisor IOS

```
Switch(config)#ip access-list extended ip_subnet2host
Switch(config-ext-acl)#permit ip 10.101.0.0 0.0.255.255 host
10.101.1.1
Switch(config)#ip access-list extended ping
Switch(config-ext-acl)#permit icmp any any echo
Switch(config-ext-acl)#permit icmp any any echo-reply
Switch(config-ext-acl)#exit
Switch(config)#ip access-list extended_icmp
Switch(config-ext-acl)#permit icmp any any
Switch(config-ext-acl)#exit
Switch(config)#ip access-list extended_tcp
Switch(config-ext-acl)#permit tcp any any
Switch(config-ext-acl)#exit
Switch(config)#ip access-list extended_udp
Switch(config-ext-acl)#permit udp any any
Switch(config-ext-acl)#exit
Switch(config)#vlan access-map watchlist
Switch(config-access-map)#match ip address ip_subnet2host
Switch(config-access-map)#action forward
Switch(config-access-map)#vlan access-map watchlist 10
Switch(config-access-map)#match ip address ping
Switch(config-access-map)#action forward
Switch(config-access-map)#vlan access-map watchlist 20
Switch(config access-map)#match ip address ip_icmp
Switch(config-access-map)#action drop
Switch(config-access-map)#vlan access-map watchlist 30
Switch(config-access-map)#match ip address ip_tcp
Switch(config-access-map)#action forward
Switch(config-access-map)#vlan access-map watchlist 40
Switch(config-access-map)#match ip address ip_udp
Switch(config-access-map)#action drop
Switch(config-access-map)#vlan access-map watchlist 50
Switch(config access-map)#action forward
Switch(config-access-map)#exit
Switch(config)#vlan filter watchlist vlan-list 101
Switch(config)#end
Switch(config)#copy running-config startup-config
```

11.5: аутентификация на коммутаторе

- Функция аутентификации на коммутаторе позволяет контролировать доступ пользователей к устройству.
- Стандартно аутентификация контролируется локально с помощью пользовательского пароля и пароля привилегированного уровня.
- Существует возможность настроить коммутатор на использование сервера аутентификации, такого, как RADIUS или TACACS+.
- После того как протоколы RADIUS и TACACS+ настроены, важно включить локальную аутентификацию для регистрации в коммутаторе в случае, если сервер аутентификации отключен.
- В некоторых случаях настройка аутентификации требуется для таких функций, как SSH (*Secure Shell*) Telnet и аутентификация порта стандарта 802.1X.

Конфигурирование функции

Аутентификация на коммутаторе определяет порядок проверки пользователей перед предоставлением им доступа к пользовательскому или привилегированному интерфейсу командной строки. Аутентификация может быть настроена на использование локальных паролей на коммутаторе или таким образом, что пользователи проходят авторизацию на сервере TACACS или RADIUS. Для контроля аутентификации пользователей на коммутаторе используются описанные ниже команды.

1. Конфигурирование локальной аутентификации.

   В стандартной аутентификации используются пароли на коммутаторе. Команды, перечисленные в этом разделе, показывают, каким образом включать или отключать стандартную аутентификацию. Даже если для аутентификации используется какой-либо сервер, локальную аутентификацию отключать не следует, поскольку она обеспечивает "черный ход" (back door) — дополнительную возможность аутентификации при отказе сервера. В коммутаторе имеются два уровня аутентификации: пользовательский и привилегированный. Команды, приведенные ниже, показывают, как управлять аутентификацией для каждого уровня.

   а) Конфигурирование аутентификации пользовательского уровня.

Система COS	`set authentication login local {enable | disable} [all | console | telnet | http]`

   Эта команда используется для включения и отключения локальной аутентификации пользовательского уровня для таких служб, как `console`, `telnet`, `http` и все (`all`) службы COS-коммутатора.

   б) Конфигурирование аутентификации привилегированного уровня.

Система COS	`set authentication enable local {enable | disable} [all | console | telnet | http]`

   Приведенная выше команда применяется для включения и отключения локальной аутентификации привилегированного уровня для таких служб, как `console`, `telnet`, `http` и все (`all`) службы COS-коммутатора.

2. Конфигурирование TACACS-аутентификации.

Существует также возможность настроить аутентификацию пользователей из базы данных TACACS-сервера. Чтобы использовать такую возможность, необходимо сконфигурировать на TACACS-сервере имя пользователя и пароль. После конфигурирования сервера для обеспечения TACACS-аутентификации используются описанные ниже команды.

а) Конфигурирование TACACS-сервера.

Система COS	`set tacacs server` *address* `[primary]`

Эта команда определяет адрес TACACS-сервера. При этом предполагается, что компьютеру задан IP-адрес, а также имеется порт, необходимый для связи с данным сервером. Можно указать несколько серверов на случай, если одно из устройств не будет функционировать. Параметр `primary` определяет, какой сервер опрашивается в первую очередь.

б) Включение TACACS-аутентификации для пользовательского уровня.

Система COS	`set authentication login tacacs {enable | disable} [all | console | telnet | http]`

После того как адрес сервера задан, процесс аутентификации пользовательского уровня устанавливается на использование параметра `tacacs` для доступа к службам `console`, `telnet`, `http` или всем службам (`all`). Параметр `primary` для этой команды указывает на то, что алгоритм TACACS является первым методом аутентификации и в случае, если он терпит неудачу, применяются другие методы, такие, как локальная регистрация.

в) Включение TACACS-аутентификации для привилегированного уровня.

Система COS	`set authentication enable tacacs {enable | disable} [all | console | telnet | http]`

После того как адрес сервера задан, процесс аутентификации привилегированного уровня устанавливается на использование параметра `tacacs` для доступа к службам `console`, `telnet`, `http` или всем службам (`all`). Параметр `primary` для этой команды указывает на то, что алгоритм TACACS является первым методом аутентификации и в случае, если он терпит неудачу, применяются другие методы, такие, как локальная регистрация.

г) Указание TACACS-ключа.

Система COS	`set tacacs key` *key*

Поскольку передаваемая между TACACS-устройством и коммутатором информация шифруется, необходимо также сообщить TACACS-процессу ключ (*key*), который используется сервером. Приведенная выше команда указывает используемый ключ.

3. Конфигурирование RADIUS-аутентификации.

В дополнение к локальной, или TACACS-аутентификации, можно настроить коммутатор на аутентификацию пользователей из базы данных RADIUS-сервера. Чтобы использовать такую возможность, необходимо сконфигурировать на RADIUS-

сервере имя пользователя и пароль. После конфигурирования сервера для обеспечения RADIUS-аутентификации используются описанные ниже команды.

а) Конфигурирование RADIUS-сервера.

Система COS	`set radius server address [auth-port port] [primary]`

Эта команда задает адрес RADIUS-сервера. При этом предполагается, что коммутатору задан IP-адрес, а также имеется шлюз, необходимый для связи с сервером. Можно указать несколько серверов на случай, если одно из устройств не будет функционировать. Параметр **primary** определяет, какой сервер опрашивается в первую очередь.

б) Включение RADIUS-аутентификации для пользовательского уровня.

Система COS	`set authentication login radius {enable | disable} [all | console | telnet | http]`

После того как адрес сервера задан, процесс аутентификации пользовательского уровня устанавливается на использование параметра `radius` для доступа к службам `console`, `telnet`, `http` или всем службам (`all`). Параметр **primary** для этой команды указывает на то, что алгоритм RADIUS является первым методом аутентификации и в случае, если он терпит неудачу, применяются другие методы, такие, как локальная регистрация.

в) Включение RADIUS-аутентификации для привилегированного уровня.

Система COS	`set authentication enable radius {enable | disable} [all | console | telnet | http]`

После того как адрес сервера задан, процесс аутентификации привилегированного уровня устанавливается на использование параметра `radius` для доступа к службам `console`, `telnet`, `http` или всем службам (`all`). Параметр **primary** для этой команды указывает на то, что алгоритм RADIUS является первым методом аутентификации и в случае, если он терпит неудачу, применяются другие методы, такие, как локальная регистрация.

г) Указание RADIUS-ключа.

Система COS	`set radius key` *key*

Поскольку информация, передаваемая между RADIUS-устройством и коммутатором, шифруется, необходимо также сообщить RADIUS-процессу ключ (Key), который используется сервером. Приведенная выше команда определяет используемый ключ.

Проверка конфигурации

Для проверки настроек аутентификации используются перечисленные ниже команды.

Система COS	`show authentication`
	`show tacacs`
	`show radius`

Пример конфигурирования функции

В этом примере демонстрируется конфигурация коммутатора, использующего RADIUS-сервер с адресом 192.168.1.10 в качестве главного метода аутентификации для Telnet-пользователей, а также TACACS-сервер с адресом 192.168.1.8 как главный метод аутентификации для пользователей консольных служб. TACACS-ключ — abc123, RADIUS-ключ — 789xyz.

Ниже приведен пример конфигурации Catalyst OS.

```
Catalyst (enable)> set radius server 192.168.1.10
Catalyst (enable)> set authentication login radius enable telnet primary
Catalyst (enable)> set authentication enable radius enable telnet primary
Catalyst (enable)> set radius key 789xyz
Catalyst (enable)> set tacacs server 192.168.1.8
Catalyst (enable)> set authentication login tacacs enable console primary
Catalyst (enable)> set authentication enable tacacs enable console primary
Catalyst (enable)> set tacacs key abc123
```

11.6: списки разрешения доступа

- Списки разрешений доступа (Permit lists) используются в COS-коммутаторах для определения устройств, которым разрешен доступ к коммутатору по протоколам Telnet, HTTP и SNMP.
- Можно сконфигурировать записи в списке так, чтобы они применялись к SNMP- или Telnet-доступу, либо создать общий список для обоих протоколов.
- В список разрешения доступа можно ввести до ста адресов.
- Записи в списках сравниваются с инвертированной маской. Если маска не указана, сравниваются все биты адреса.
- Списки разрешения доступа не оказывают влияния на внешний Telnet-трафик и административный трафик.
- При обнаружении попытки неавторизованного доступа возможна генерация SNMP-прерываний.

Конфигурирование функции

Для конфигурирования списка разрешения доступа на COS-коммутаторе используются описанные ниже команды.

1. Добавление адресов в список разрешений.

Система COS	`set ip permit` *address mask* `[snmp | telnet]`

Чтобы контролировать, каким устройствам разрешен доступ к коммутатору, прежде всего необходимо сконфигурировать список разрешения IP-доступа. Параметр *address* определяет IP-адрес устройства, которому разрешен доступ к сети. Параметр *mask* использовать необязательно. Маска задается в точечно-десятичном представлении, где единица означает совпадение с адресом, а ноль означает игнорирование адреса. Например, адрес 172.16.101.1 с маской 255.255.255.0 соответствует всем

адресам, которые начинаются с 172.16.101. Тот же адрес с маской 255.255.255.255 соответствует только одному узлу — 172.16.101.1. В случае, если маска не задана, используется маска, в которой все биты равны единице, или маска узла. Параметры **snmp** и **telnet** задают процесс, который будет использовать определенную запись списка. Если процесс не задан, запись применяется ко всем процессам.

2. Активизация списка разрешения доступа.

Система COS	`set ip permit enable [snmp | telnet]`

После того как будет сконфигурирован список устройств, которым разрешен доступ, для включения списка разрешения доступа используется приведенная выше команда. Параметры **snmp** и **telnet** задают процесс, для которого активизируется список разрешения доступа.

3. Включение генерации SNMP-прерываний (*необязательно*).

Система COS	`set snmp trap enable ip permit`

Эта команда активизирует процесс `ip permit` для отправки SNMP-прерываний в случае обнаружения попытки неавторизованного доступа к коммутатору.

Проверка конфигурации

Для проверки конфигурации списка IP-доступа используется команда **show ip permit**.

Система COS	`show ip permit`

Пример конфигурирования функции

В этом примере демонстрируется конфигурация списка разрешения доступа. Такой список позволяет любому пользователю из сети 192.168.5.0 получить доступ к SNMP- и Telnet-службам конфигурируемого устройства. Кроме того, рассматриваемая конфигурация позволяет всем пользователям из подсети 192.168.1.0 получить доступ к устройству по протоколу Telnet. В дополнение к этому наш пример содержит запись, которая позволяет узлу 192.168.255.1 связываться с устройством посредством SNMP. Этот список также активизируется для протоколов Telnet и SNMP.

Ниже приводится пример конфигурации Catalyst OS.

```
Console (enable)>set ip permit 192.168.5.0 255.255.255.0
Console (enable)>set ip permit 192.168.1.0 255.255.255.0 telnet
Console (enable)>set ip permit 192.168.255.1 255.255.255.255 snmp
Console (enable)>set ip permit telnet
Console (enable)>set ip permit snmp
```

Совет

При создании списка разрешения IP-доступа прежде всего следует внести адрес станции управления. Это необходимо для того, чтобы предотвратить блокирование доступа к коммутатору для администратора.

11.7: конфигурация служб SSH и Telnet

- Telnet-подключения к коммутатору производятся по протоколу TCP порт 23, а данные передаются в виде простого текста.
- Если какой-либо пользователь, использующий сетевой анализатор, перехватит пакеты, следующие к серверу, то он получит возможность увидеть данные, передаваемые в виде простого текста, включая пароли.
- Технология SSH (Secure Shell — безопасное удаленное соединение) представляет собой метод обмена данными по протоколу Telnet, при котором пакеты перед транспортировкой между устройствами шифруются.
- Протокол SSH выполняется на TCP-порту 22 между SSH-совместимым клиентом и устройством, настроенным на прием SSH-соединений.
- Коммутаторы корпорации Cisco поддерживают только SSH версии 1.
- Чтобы реализовать SSH на конкретном коммутаторе, необходимо установить на нем программное обеспечение с поддержкой криптографии.
- По умолчанию на коммутаторе протокол SSH отключен, и его необходимо активизировать, прежде чем клиенты получат возможность подключиться.

Конфигурирование функций

Для обеспечения безопасной Telnet-связи между коммутатором и SSH Telnet-клиентом необходимо разрешить на коммутаторе SSH-связь. Ниже приводятся конфигурационные этапы активации SSH.

1. Установка кодированного ключа.

Система COS	`set crypto key rsa number`
Система IOS	`crypto key generate rsa` (режим глобальной конфигурации)

Прежде чем появится возможность конфигурировать службу SSH, необходимо разрешить коммутатору генерировать ключ для шифрования данных. Ключ генерируется с помощью команды `crypto key rsa`. В COS-коммутаторах параметр *option* определяет предельную длину (modulus length). В системе IOS это значение запрашивается. Чем больше длина, тем более строгое шифрование используется. Рекомендуемая предельная величина равна 1024 или более.

2. Указание устройства, которому разрешено использовать SSH.

Система COS	`set ip permit address mask ssh`
Система IOS	Нет

Для COS-коммутаторов SSH включается с помощью списков разрешений IP-доступа. Для активизации этого процесса в первую очередь необходимо указать устройства, которым разрешен доступ к коммутатору с использованием SSH. Если указывать определенный адрес или диапазон адресов неже-

дательно, то следует ввести адрес с маской 0.0.0.0, что позволит использовать SSH-процесс всем устройствам.

3. Включение списка разрешений SSH-доступа.

Система COS	`set ip permit enable ssh`
Система IOS	`ip ssh` (режим глобальной конфигурации)

Для фактической активизации SSH-процесса в COS-коммутаторах используется команда `set ip permit enable ssh`. Она позволяет подключить к SSH-процессу IP-адреса, определенные на этапе 2. В IOS-коммутаторах SSH включается с помощью глобальной команды `ip ssh`.

Проверка конфигурации

Для проверки конфигурации SSH используются команды, перечисленные ниже:

Система IOS	`show ip ssh` `show ip permit`

Пример конфигурирования функции

В этом примере отражена конфигурация, позволяющая любому устройству получить доступ к коммутатору с помощью протокола SSH. Предельная длина RSA-ключа для данного коммутатора устанавливается равной 1024.

Ниже приводится пример конфигурации Catalyst OS.

```
Catalyst (enable)>set crypto key rsa 1024
Catalyst (enable)>set ip permit 0.0.0.0 0.0.0.0 ssh
Catalyst (enable)>set ip permit ssh
```

Пример конфигурации для операционной системы Supervisor IOS.

```
Switch(config)#crypto key generate rsa
Enter modulus:1024
Switch(config)#ip ssh
Switch(config)#end
Switch(config)#copy running-config startup-config
```

11.8: аутентификация по протоколу 802.1X

- В большинстве коммутаторов порты активизируются по умолчанию, и любой пользователь, который может подключиться к порту, получает доступ к сети.
- Функция безопасности портов, использующая MAC-адреса, способна управлять доступом устройств к сети на защищенном порту, но требует переконфигурирования при перемещении устройства.
- Спецификация 802.1X обеспечивает стандартный метод для авторизации портов с помощью клиентских сертификатов или имен пользователей.
- Для обеспечения авторизации используемого порта в стандарте 802.1X используется RADIUS-сервер.

- До тех пор пока порт 802.1X не авторизован, он не может использоваться для передачи пользовательского трафика.
- В стандарте 802.1X коммутатор функционирует как прокси (proxy) между клиентом и сервером для передачи сведений аутентификации.

Конфигурирование функции

Ниже описаны этапы конфигурирования аутентификации порта 802.1X.

1. Глобальное включение 802.1X-аутентификации

Система COS	`set dot1x system-auth-control enable`
Система IOS	Нет

В COS-коммутаторе в первую очередь необходимо глобально включить процесс 802.1X-аутентификации, прежде чем появится возможность конфигурировать порты для авторизации.

2. Указание RADIUS-сервера и ключа.

Система COS	`set radius server` *address* `set radius key` *string*
Система IOS	`radius-server host` *address* `key` *string* (режим глобальной конфигурации)

Поскольку 802.1X-процесс зависит от RADIUS-сервера, в коммутаторе необходимо задать адрес сервера и используемый им ключ.

3. Создание модели AAA (*Authentication, Authorization, Accounting — аутентификация, авторизация, учет*).

Система COS	Нет

Система IOS	`aaa new-model`
	`aaa authentication dot1x default group radius`
	(обе команды вводятся в режиме глобальной конфигурации)

В IOS-коммутаторе 802.1X-аутентификация включается путем создания AAA-модели с помощью вышеперечисленных команд.

4. Включение на порту протокола 802.1X.

Система COS	`set port dot1x` *mod/port* `port-control auto`
Система IOS	`dot1x port-control {auto | force-authorized | force-unauthorized}` (режим конфигурирования интерфейса)

После завершения предыдущих этапов можно настроить порт для 802.1X-авторизации. Порт, сконфигурированный для 802.1X-аутентификации, не передает пользовательский трафик до тех пор, пока RADIUS-сервер не отправит сведения авторизации для данного порта.

Пример конфигурирования функции

В этом примере демонстрируется конфигурация Ethernet-порта 3/6 для обеспечения 802.1X-аутентификации клиента с помощью RADIUS-сервера 10.1.1.1 со строкой ключа *funhouse*.

Ниже приведен пример конфигурации Catalyst OS.

```
Catalyst (enable)set dot1x system-auth-control enable
Catalyst (enable)set radius server 10.1.1.1
Catalyst (enable)set radius key funhouse
Catalyst (enable)set port dot1x 3/6 port-control auto
```

Ниже приведен пример конфигурации Supervisor IOS.

```
Switch(config)#radius-server host 10.1.1.1 key funhouse
Switch(config)#aaa new-model
Switch(config)#aaa authentication dot1x default group radius
Switch(config)#interface fastethernet 3/6
Switch(config-if)#dot1x port-control auto
Switch(config-if)#end
Switch(config)#copy running-config startup-config
```

Дополнительная литература

Рекомендуемые ниже источники предоставляют более подробную информацию по темам, рассматриваемым в этой главе.

Кеннеди Кларк, Кевин Гамильтон. *Принципы коммутации в локальных сетях Cisco*, ИД "Вильямс", 2003.

Karen Webb. *Building Cisco Multilayer Switched Networks*, Cisco Press.

Tim Boyles and David Hucaby. *CCNP Switching Exam Certification Guide*. Cisco Press.

David Hucaby Steve McQuerry. *Cisco Field Manual: Router Configuration*. Cisco Press.

В этой главе...

- 12.1: протоколирование событий. В разделе описываются этапы конфигурирования различных методов протоколирования сообщений, поступающих от коммутатора.
- 12.2: простой протокол управления сетью. В разделе представлена информация о конфигурировании коммутатора для использования протоколов сетевого администрирования.
- 12.3: анализатор коммутируемых портов. В этом разделе описаны настройки, позволяющие отражать трафик коммутатора для сетевого анализа либо локально, либо на удалённом коммутаторе.
- 12.4: управление питанием. В разделе рассматриваются команды коммутатора Catalyst 6000 для управления питанием шасси и модулей.
- 12.5: мониторинг температуры. В разделе рассматриваются команды коммутатора Catalyst 6000 для отображения информации о температуре коммутатора.
- 12.6: трассировка пакетов. В разделе описываются некоторые методы трассировки пакетов второго и третьего уровней в сети. С любого коммутатора можно протестировать связь с удалённым узлом.

Глава 12

Управление коммутаторами

12.1: протоколирование событий

- Протоколирование (logging) используется коммутатором для отправки системных сообщений протоколирующей системе.
- Протоколируемые сообщения могут быть отправлены через любое из четырех различных средств: консольный порт коммутатора, файл на коммутаторе, Telnet-сеанс или сервер syslog.
- В файле может сохраняться история протоколирования, которая необходима для обеспечения сохранности записи сообщений, отправляемых сервером SNMP или syslog, на случай, если будет утерян или удален какой-либо пакет.
- Протоколирование отражает все ошибки и отладочные сообщения по умолчанию. Чтобы определить, какие сообщения следует отправлять на каждое устройство, можно устанавливать уровни протоколирования.
- Временные метки протокольных сообщений или установка адреса syslog-отправителя могут помочь при отладке и администрировании в реальном времени. Если время и дата устанавливаются на коммутаторе, то он способен обеспечить временные метки для каждого syslog-сообщения. Возможна синхронизация аппаратных часов во всех коммутаторах, в результате чего упрощается сопоставление syslog-сообщений от нескольких устройств.

Системные сообщения протоколируются в следующем формате:

```
timestamp %function-severity-MNEMONIC:description,
```

где параметр `timestamp` (временная метка) отмечает время события, а параметр `%function` описывает функцию коммутатора (также называется средством), генерирующей это событие, параметр `severity` представляет собой уровень важности протоколирования (от 0 до 7; чем ниже уровень, тем важнее сообщение) события, значение параметра `MNEMONIC` — текстовая строка, с помощью которой кратко описывается событие. Завершает сообщение текстовая строка, содержащая более подробное описание сообщения — `description`.

Ниже приводится пример системного сообщения Supervisor IOS третьего уровня важности.

```
1w1d: %LINK-3-UPDOWN: Interface FastEthernet5/10, changed state to up
```

Ниже приводится пример системного сообщения Catalyst OS пятого уровня важности.

```
2001 Dec 20 11:40:19 %DTP-5-NONTRUNKPORTON:Port 5/4 has become non-trunk
```

Внимание!

При протоколировании на syslog-сервер используется UDP-порт 514.

Конфигурирование функции

1. Включение или отключение протоколирования *(необязательно)*.

Система COS	**Нет**
Система IOS	(no) **logging on** (режим глобальной конфигурации)

Функция протоколирования стандартно включена. Чтобы отключить на коммутаторе все протоколирование, кроме протоколирования на консоль, используется ключевое слово no.

2. Протоколирование сообщений на syslog-сервер *(необязательно)*.

   а) Указание syslog-сервера.

Система COS	**set logging server** *syslog-host*
Система IOS	**logging** *syslog-host* (режим глобальной конфигурации)

Текстовые сообщения отправляются syslog-серверу с адресом *syslog-host* (имя узла или IP-адрес). Сообщения перехватываются и могут быть просмотрены на syslog-сервере.

   б) Отправка сообщений syslog-средству.

Система COS	**set logging server facility** *facility-type*
Система IOS	**logging facility** *facility-type* (режим глобальной конфигурации)

Когда syslog-сервер получает сообщение, то перенаправляет его в log-файл или получателю в зависимости от типа протоколирующего средства исходной системы. В таком режиме серверы syslog могут накапливать и организовывать сообщения, используя данное средство в качестве объекта или типа обслуживания. Если средство протоколирования установлено идентично на всех коммутаторах, то все syslog-сообщения от коммутаторов могут быть собраны вместе.

Syslog-серверы основаны на концепциях операционной системы UNIX и имеют средства, которые названы именами различных системных служб. Средство, используемое в syslog-сообщениях коммутатора, определяется параметром *facility-type*, который может принимать одно из восьми значений: `local0`, `local1`, `local2`, `local3`, `local4`, `local5`, `local6`, `local7` (стандартное значение). Все указанные типы соответствуют локально определенным службам. Обычно для сообщений от сетевых устройств используется одно или несколько локальных средств. Операционная система Supervisor IOS также допус-

кает использование следующих дополнительных типов средств: auth (службы аутентификации пользователей), cron (службы планировки задач), daemon (фоновые системные службы), kern (службы системного ядра), lpr (службы спулера принтера), mail (службы системной почты), news (службы групп новостей Usenet), syslog (syslog-службы), sys9, sys10, sys11, sys12, sys13, sys14 (все зарезервированы для системных служб), user (процессы системного пользователя) или uucp (службы копирования файлов UNIX-to-UNIX)

в) Ограничение по важности протоколируемых сообщений

Система COS	`set logging server severity level`
Система IOS	`logging trap level` (режим глобальной конфигурации)

Системным сообщениям присваивается уровень важности на основании типа и важности условий возникновения ошибки. Syslog-серверу отправляются только сообщения, которые имеют важность ниже или равную (по крайней мере так же важны) установленному уровню важности. Параметр *level* — число (от 0 до 7; стандартно 6), определенное в табл. 12.1

Таблица 12.1. Уровни важности системных сообщений

Уровень	Название	Описание
0	`emergencies`	Систему невозможно использовать
1	`alerts`	Необходимо немедленное вмешательство
2	`critical`	Возникла критическая ошибка
3	`errors`	Возникла ошибка
4	`warnings`	Условия, требующие внимания
5	`notifications`	Нормальное состояние, которое вызывает появление системного сообщения
6	`informational`	Информационные сообщения
7	`debugging`	Отладочные сообщения

Операционная система IOS также позволяет вводить уровень (level) как имя. Большинство изменений физического состояния (включение или выключение портов и модулей) протоколируются на уровне 5, тогда как аппаратные и программные ошибки протоколируются на уровне 3.

г) Использование определенного адреса отправителя для syslog-сообщений (*необязательно; только для IOS-коммутаторов*).

Система COS	Нет
Система IOS	`logging source-interface type number` (режим глобальной конфигурации)

IOS-коммутатор может использовать в syslog-пакетах IP-адрес определенного интерфейса в качестве адреса отправителя. Такая функция может быть полезна в ситуации, когда имеется множество интерфейсов, однако необходимо просматри-

ять все syslog-сообщения от коммутатора как поступившие с одного адреса. С той же целью в COS коммутаторе всегда используется адрес SC0-интерфейса.

а) Включение протоколирования с использованием syslog-сервера (*только для OS коммутаторов*).

Система COS	`set logging server enable`
Система IOS	Нет

с) Ограничение количества записываемых в таблицу SNMP журнала сообщений (*необязательно, только для IOS-коммутаторов*).

Система COS	Нет
Система IOS	`logging history` *level* `logging history size` *number* (обе команды вводятся в режиме глобальной конфигурации)

Сообщения, отправляемые администрацией SNMP-станцией в виде прерываний, могут быть потеряны. Отсюда сообщения, имеющие уровень важности ниже или равный указанному, могут также сохраняться в таблице истории для последующего просмотра. Параметр *level* представляет собой число от 0 до 7, определенное в табл. 12.1. Стандартно только одно сообщение хранится в таблице истории. Эту установку можно изменить, используя ключевое слово `size` с параметром *number* (число записей сообщений от 1 до 500).

3. Протоколирование сообщений в буфер коммутатора (*необязательно*)

Система COS	`set logging buffer` *size*
Система IOS	`logging buffered` *[size]* (режим глобальной конфигурации)

Все системные сообщения сохраняются в какой-либо области памяти коммутатора. Целостность буфера сообщений сохраняется до тех пор, пока коммутатор не будет выключен или буфер не будет очищен с помощью команды `clear logging`. Максимальный размер буфера может быть задан параметром *size* (COS: от 1 до 500 сообщений, стандартно 500; система Supervisor IOS: 4096 до 2147483647 байтов, стандартно 4096 байтов).

Внимание!

Размер буфера изменяется в зависимости от коммутирующих платформ Catalyst. В коммутаторе с Supervisor IOS протоколирование в буфер использует системные ресурсы, которые также могут быть необходимы для функциональных задач коммутатора. Следует разумно подходить к установке максимального размера буфера с тем, чтобы бесполезно не расходовать системную память.

4. Протоколирование сообщений в файл на коммутаторе (*необязательно, только для IOS коммутаторов*)

Система COS	Нет
Система IOS	`logging file [flash:]`*filename [max-file-size] level* (режим глобальной конфигурации)

Системные сообщения хранятся в файле с именем *filename* (текстовая строка), расположенном в системном устройстве `disk0:`. Размер файла можно ограничить с помощью параметров *max file size* (максимальный размер, от 4096 до 2147483647 байтов, стандартно 4096) и *min file size* (минимальный размер, от 1024 до 2147483647 байтов, стандартно 2048). В этот файл добавляются сообщения с уровнем важности меньшим или равным заданному уровню (параметр *level* от 0 до 7 или имя из табл. 12.1, стандартное значение равно 7 или `debugging`).

5. Запись сообщений в терминальные сеансы (*необязательно*)

а) Вывод сообщений на консоль коммутатора (*необязательно*).

Система COS	`set logging console {enable | disable}`
Система IOS	`logging console` *level* (режим глобальной конфигурации)

Стандартно системные сообщения выводятся на консоль. Отключить функцию протоколирования можно с помощью ключевого слова `disable`. В IOS-коммутаторах на консоль отправляются только сообщения с уровнем важности меньшим или равным заданному (параметр *level* от 0 до 7 или имя из табл. 12.1, стандартное значение равно 7 или `debugging`).

б) Запись сообщений в Telnet-сеанс или сеанс линии (*необязательно*).

Система COS	`set logging telnet {enable | disable}`
Система IOS	`logging monitor` *level* (режим глобальной конфигурации)

Стандартно системные сообщения протоколируются во всех Telnet-сеансах или сеансах терминальных линий. В COS-коммутаторах можно также отключить протоколирование, но только для текущего сеанса, используя команду `set logging session disable`. В IOS-коммутаторах в сеанс отправляются только сообщения с уровнем важности меньшим или равным заданному (параметр *level* от 0 до 7 или имя из табл. 12.1, стандартно 7 или `debugging`).

Внимание!

Для просмотра системных сообщений в течение Telnet-сеанса (соединения vty) в IOS-коммутаторе необходимо использовать команду EXEC-режима `terminal monitor`.

в) Управление выводом сообщений на терминальные сеансы (*необязательно; только для IOS-коммутаторов*).

Система COS	Нет
Система IOS	`logging synchronous [level` *level* `| all] [limit` *buffers*`]` (режим конфигурирования линии)

В ситуациях, когда используется синхронное протоколирование, сообщения устанавливаются в очередь до тех пор, пока не будет отображена затребован-

ная информация (например, обычный вывод от команды show или конфигурационной команды). Вывод протокола будет отображен в момент появления командной строки. Синхронизация может использоваться для сообщений, которые имеют уровень важности ниже или равный указанному (параметр level от 0 до 7 или имя из табл. 12.1, стандартно 2), либо все уровни (all). С помощью ключевого слова limit можно заставить коммутатор устанавливать в очередь определенное число сообщений (параметр buffered, стандартно 20), прежде чем они будут удалены из очереди.

Совет

Несмотря на то что синхронное протоколирование позволяет отделить сообщение коммутатора при вводе или чтении другого отображаемого текста, оно также может привести к путанице. Например, в ситуации, когда включено синхронное протоколирование на консоль коммутатора и ни один из пользователей в этот момент не зарегистрирован в системе, коммутатор будет устанавливать в очередь все сообщения до тех пор, пока в системе не зарегистрируется пользователь. Этот пользователь увидит огромное количество сообщений, которые будут постранично отображаться, включая информацию, возможно, зарегистрированную несколькими часами или днями ранее.

6. Запись временной метки в каждое системное сообщение (*необязательно*).

Система COS	`set logging timestamp {enable | disable}`
Система IOS	`service timestamps log {uptime | datetime}` (режим глобальной конфигурации)

Стандартно вместе с системными сообщениями COS-коммутаторы записывают временную метку с указанием даты и времени, тогда как IOS-коммутаторы записывают время работы системы. Для того чтобы использовать дату и время, следует применить ключевое слово `datetime`. Это может оказаться полезным, если необходимо определить фактическое время возникновения ошибочных условий.

Совет

Прежде чем полагаться на временные метки протоколируемых сообщений, на коммутаторе следует сконфигурировать и установить точное время, дату и часовой пояс. Более подробная информация по этому вопросу приведена в разделе "3.8: установка времени и даты".

7. Управление скоростью создания системных сообщений (*необязательно; только для IOS-коммутаторов*).

Система COS	Нет
Система IOS	`logging rate-limit` *number* `{all | console} {except level}` (режим глобальной конфигурации)

Чтобы предотвратить отправку чрезмерного количества системных сообщений протоколирующему получателю, следует ограничить скорость, с которой отправляются сообщения (параметр *number* от 1 до 10000 в секунду, стандартного значения нет). Ключевое слово `all` позволяет ограничить все сообщения, тогда как ключевое слово `console` ограничивает только сообщения, отправляемые на консоль коммутатора. Можно использовать ключевое слово `except` для ограни-

ченных сообщений с уровнем важности ниже или равным указанному (параметр *level*) от 0 до 7, значения определены в табл. 12.1).

8. Установка уровня важности для определенных функций коммутатора (*необязательно; только для COS-коммутаторов*).

Система COS	`set logging level {all | function} level [default]`
Система IOS	Нет

Системные сообщения могут быть занесены в журнал, если их уровень важности равен или ниже установленного. Кроме того, можно точно настроить уровень важности определенных функций коммутатора, для того чтобы определять, какие будут регистрироваться и будут ли регистрироваться вообще их сообщения. Можно использовать ключевое слово `all` для того, чтобы установить один уровень для всех функций коммутатора (параметр *level* от 0 до 7 или имя, определенное в табл. 12.1).

В противном случае уровень может быть назначен одной из следующих функций: `acl` (списки контроля доступа), `cdp` (*Cisco Discovery Protocol — протокол обнаружения устройств Cisco*), `copa` (*Common Open Policy Server — общий открытый сервер правил*), `dtp` (*Dynamic Trunking Protocol — динамический магистральный протокол*), `dvlan` (динамические VLAN-сети), `earl` (*Enhanced Address Recognition Logic — логика улучшенного распознавания адресов*), `filesys` (файловые системы), `gvrp` (*GARP VLAN Registration Protocol — GARP-протокол регистрации сетей VLAN*), `ip` (Internet-протокол), `kernel` (ядро коммутатора), `ld` (*Accelerated Server Load Balancing, ASLB — ускоренная балансировка нагрузки на серверы*), `mcast` (многоадресатное вещание), `mgmt` (административные функции), `mls` (*Multi Layer Switching, MLS — многоуровневая коммутация*), `pagp` (*Port Aggregation Protocol, PagP — протокол суммирования портов*), `protfilt` (фильтрация протоколов), `pruning` (отсекание VLAN сетей), `privatevlan` (частные виртуальные локальные сети), `qos` (*Quality of Service, QoS — качество обслуживания*), `radius` (*Remote Access Dial In User Service — служба идентификации удаленных абонентов*), `rsvp` (*Resource Reservation Protocol — протокол резервирования ресурсов*), `security` (функции обеспечения безопасности), `snmp` (*Simple Network Management Protocol, SNMP — простой протокол управления сетью*), `spantree` (*Spanning Tree Protocol, STP — протокол распределенного связующего дерева*), `sys` (системные функции), `tac` (*Terminal Access Controller — контроллер терминального доступа*), `tcp` (*Transmission Control Protocol, TCP — протокол управления передачей*), `telnet` (*Terminal Emulation Protocol Telnet — протокол эмуляции терминала*), `tftp` (*Trivial File Transfer Protocol, TFTP — простой протокол передачи файлов*), `udld` (*Unidirectional Link Detection — однонаправленное обнаружение канала*), `vmps` (*VLAN Membership Policy Server — сервер прав VLAN-сетей*) или `vtp` (*VLAN Trunking Protocol, VTP — протокол магистральных каналов VLAN сетей*).

Стандартно в COS-коммутаторах этим функциям назначены следующие уровни важности: `sys` (5), `dtp` (5), `pagp` (5), `mgmt` (5), `mls` (5), `cdp` (4), `udld` (4), `ip` (3), `qos` (3) и все остальные функции (2).

Уровни важности модифицируются только для текущего сеанса. Чтобы модифицировать уровни всех сеансов, следует использовать ключевое слово `default`.

Пример конфигурирования функции протоколирования

В этом примере настраивается функция протоколирования сообщений коммутатора на syslog сервер с адресом 192.168.254.91. Стандартно используется средство local7 с сообщениями, уровень которых равен 6, т.е. информационные сообщения, или ниже. COS-коммутатор настраивается на запись до пяти сообщений в свой внутренний буфер, тогда как IOS-коммутатор записывает в буфер до 64 Кбайт текста сообщений. В COS-коммутаторе отключен вывод системных сообщений в текущий Telnet сеанс.

Коммутатор предоставляет временные метки даты и времени перед каждым протоколируемым сообщением. COS-коммутатор также имеет дополнительную настройку VTP сообщения протоколируются, если уровень их важности равен пяти (извещения) или ниже.

Система COS	`set logging server 192.168.254.91`
	`set logging server enable`
	`set logging buffer 500`
	`set logging session disable`
	`set logging timestamp enable`
	`set logging level vtp 5 default`

Система IOS	`logging 192.168.254.91`
	`logging buffered 65536`
	`service timestamps log datetime`
	(все команды вводятся в режиме глобальной конфигурации)

Отображение информации о функции протоколирования

В табл. 12.2 перечислены некоторые команды коммутатора, которые можно использовать для отображения полезных сведений о протоколировании системных сообщений.

Таблица 12.2. Команды коммутатора для отображения информации о функции системного протоколирования

Функция отображения	Операционная система коммутатора	Команда
Конфигурация протоколирования	COS	`show logging [noalias]`
	IOS	`show logging`
Системные сообщения	COS	`show logging buffer [-] [number of messages]`
	IOS	`show logging`

12.2: простой протокол управления сетью

- *Простой протокол управления сетью (Simple Network Management Protocol — SNMP)* — протокол, который позволяет осуществлять мониторинг информации об управлении сетевым устройством.
- *База управляющей информации (Management Information Base — MIB)* — набор переменных, хранящихся на сетевом устройстве. Переменные могут обновляться устройством и запрашиваться по желанию источника.
- MIB-объекты структурированы согласно модульному языку SNMP MIB, который основан на *языке абстрактного синтаксиса версии 1 (Abstract Syntax Notation One — ASN.1)*
- SNMP-агент выполняется на сетевом устройстве и обслуживает различные MIB-переменные. Любые обновление или запрос переменных должны обрабатываться посредством данного агента.
- SNMP-агент также способен отправлять незапрашиваемые сообщения или *прерывания (traps)* SNMP-диспетчеру. Прерывания используются для уведомления диспетчера об изменении условий на сетевом устройстве.
- SNMP-диспетчер обычно является системой управления сети, которая запрашивает MIB-переменные, может устанавливать MIB переменные и получает прерывания от группы устройств.
- SNMP-агенты способны отправлять либо прерывания, либо *информационные запросы (inform request)*. Прерывания ненадежны и отправляются в одном направлении. Информационные запросы надежны в том смысле, что они должны быть подтверждены или отправлены повторно.
- **Протокол SNMP** *версии 1* (SNMPv1) — исходная версия. Она основана на спецификации RFC 1157 и обладает только базовыми строками сообщества, предназначенными для обеспечения безопасности, которые имеют форму простого текста. Доступ также может быть ограничен по IP-адресу SNMP-диспетчера.
- **Протокол SNMP** *версии 2* (SNMPv2) представляет собой улучшенную версию, основанную на спецификациях RFC 1901, 1905 и 1906. Она надежнее работает при получении и обработке больших объемов информации и содержит усовершенствованную функцию создания отчетов об ошибках, для безопасности в ней используются текстовые строки сообществ и IP-адреса.
- **Протокол SNMP** *версии 3* (SNMPv3) основан на спецификациях RFC 2273 до 2275 и обеспечивает надежные функции безопасности. Интеграция данных и аутентификации могут обеспечиваться посредством пользовательских имен, алгоритмов *MD5 (Message Digest 5)* и *SHA (Security Hash Algorithm — алгоритм хеширования в системах безопасности)*, а также шифрования посредством стандарта *DES (Data Encryption Standard — стандарт шифрования данных)*.

Внимание!

SNMP-запросы и ответы отправляются с использованием UDP-порта 161. Информация или прерывания отправляются посредством UDP порта 162.

- *Удаленный мониторинг* (*Remote Monitoring — RMON*) позволяет просматривать трафик, проходящий через порт коммутатора. IOS-коммутаторы также могут обеспечивать RMON-предупреждения (RMON alarm) и события. RMON мониторинг поддерживает 9 административных групп, определенных в спецификации RFC 1757: статистическая (группа 1), историческая (группа 2), предупреждение (группа 3), узлы (группа 4), hostTopN (группа 5), матрица (группа 6), фильтр (группа 7), перехват (группа 8) и событие (группа 9). В поддержке RMON2 согласно спецификации RFC 2021 добавлены две группы: UsrHistory (группа 18) и ProbeConfig (группа 19).
- При включенном удаленном мониторинге коммутатор осуществляет внутреннее накопление данных. Следовательно, RMON-данные невозможно просматривать из коммутатора с помощью *интерфейса командной строки (CLI)*, они должны утилизироваться с помощью *системы управления сетью (Network Management System — NMS)*.

Конфигурирование функции

1. Конфигурирование идентификационных параметров SNMP.

   а) Указание контактной информации.

Система COS	`set system contact [contact-string]`
Система IOS	`snmp-server contact contact-string` (режим глобальной конфигурации)

   Поле `contact-string` содержит текстовую информацию о сетевом администраторе, которую может предоставить маршрутизатор. Если эта команда используется без параметра, то введенная ранее информация стирается.

   б) Указание местоположения устройства.

Система COS	`set system location [location-string]`
Система IOS	`snmp-server location location-string` (режим глобальной конфигурации)

   В поле `location-string` содержится текстовая информация, которую может предоставить маршрутизатор о своем физическом расположении. Если эта строка опущена, то введенная ранее информация стирается.

   в) Определение серийного номера устройства (только для IOS-коммутаторов).

Система COS	Нет
Система IOS	`snmp-server chassis-id id-string` (режим глобальной конфигурации)

   Параметр `id-string` — текстовая информация, предоставляемая маршрутизатором о его серийном номере. Если программное обеспечение IOS способно считать аппаратный серийный номер, то этот номер является стандартным идентификатором шасси.

2. Конфигурирование SNMP-доступа.

   а) Определение SNMP-видов для ограничения доступа к MIB-объектам *(необязательно)*.

Система COS	`set snmp view [ -hex ] {view-name} {oid-tree} [mask] [included|excluded] [volatile|nonvolatile]`
Система IOS	`snmp-server view view-name oid-tree {included|excluded}` (режим глобальной конфигурации)

При необходимости можно ограничить SNMP-диспетчер просмотром только определенных частей MIB-дерева коммутатора. Снимок (view) можно определить с помощью имени *view name*. Значение параметра *oid tree* является объектным идентификатором MIB-поддерева в формате ASN.1. Этим значением является текстовая строка с числами или словами, представляющими поддерево, разделенными с помощью точек (например, *system*, *Cisco*, *system.4.1.* 9.7*). В качестве любого компонента поддерева можно использовать маски (символ заплатки — "*"). Доступ для просмотра поддерева либо разрешен с помощью ключевого слова `included`, либо запрещен (ключевое слово `excluded`).

Можно определить множество снимков. Каждый применяется к отдельному набору пользователей или SNMP-диспетчеров.

Если имя списка содержит непечатаемые символы, то COS-коммутаторы требуют ключевое слово `-hex` и шестнадцатеричное число (*view-name*). Снимок может храниться либо в энергозависимой, либо в энергонезависимой памяти (`volatile` или `nonvolatile`). В последнем случае снимок сохраняется при выключении питания.

б) Конфигурирование методов доступа для удаленных пользователей.

- Определение строк сообщества для разрешения доступа (*SNMPv1 или SNMPv2*).

Система COS	`set snmp community {read-only | read-write |read-write-all} {string}`
Система IOS	`snmp-server community string [view view] [ro | rw] acc-list` (режим глобальной конфигурации)

Строка сообщества (значение параметра *string*) разрешает доступ к SNMP-информации на коммутаторе. Любой SNMP-диспетчер, предоставляющий соответствующую строку сообщества, получает такой доступ. Можно указать необязательный снимок с помощью ключевого слова `view` (только для IOS-коммутаторов). После этого доступ ограничивается только теми объектами, которые разрешены в определении снимка.

Доступ предоставляется в режиме "только для чтения" или "чтения-записи" с помощью ключевых слов `ro`/`read-only` (стандартное сообщество, называемое "открытым" — public, нет возможности считывать строки сообщества), `rw`/`read-write` (стандартное сообщество, называемое "закрытым" - private, может записываться любой MIB-объект, кроме строк сообщества) и `read-write-all` (стандартное сообщество, называемое "секретным" — secret, может записывать любой MIB-объект).

В IOS-коммутаторах для дальнейшего ограничения доступа только к SNMP-диспетчерам с разрешенными IP-адресами можно задать стандартный список IP-доступа с помощью необязательного параметра `acc-list`. Доступ может быть определен в SNMP-режимах "только для чтения" и "чтения-записи". В COS-коммутаторах контролироваться может только общий доступ к SNMP. Такие настройки выполняются с помощью команды `set ip permit`. В разделе "11.6: списки разрешения доступа" приведена более подробная информация о команде IP-permit.

Совет

Настоятельно рекомендуется изменить стандартные строки SNMP-сообщества на всех коммутаторах. Оставленные активными стандартные значения могут облегчить доступ неавторизованных лиц к функциям и конфигурации коммутатора. После изменения строк сообщества на уникальные значения администратору следует ограничить SNMP-доступ только IP-адресами тех узлов управления сетью, которые находятся под его контролем.

- Указание имен для идентификаторов блоков *(только для SNMPv3)*.

  Чтобы указать идентификационное имя локального блока, необходимо ввести приведенную ниже команду (команды):

Система COS	`set snmp engineid id-string`
Система IOS	`snmp-server engineID [local id-string] | [remote ip-address udp-port port id-string]` (режим глобальной конфигурации)

  В протоколе SNMP версии 3 используются аутентификация и шифрование, основанные на нескольких параметрах. Каждая сторона доверительных SNMP-отношений должна быть определена в форме идентификационной текстовой строки блока (параметр `id-string`). Этими значениями являются строки длиной в 24 символа. Могут быть указаны и более короткие строки, которые с правой стороны заполняются нулями. Локальный коммутатор, использующий протокол SNMP, должен быть определен с помощью ключевого слова `local` и параметра `id-string` (только для IOS-коммутаторов).

- Для указания имени удаленного SNMP-блока необходимо ввести следующую команду:

  `snmp-server engineID remote ip-address [udp-port port] id-string`

  Удаленный SNMP-блок (экземпляр SNMP-протокола на удаленном узле или станция управления) определяется с помощью параметра `ip-address` и имени в виде текстовой строки (`id-string`). С помощью ключевого слова `udp-port` можно задать необязательный номер UDP-порта для использования удаленным узлом (стандартное значение равно 161).

Внимание!

Если идентификационные имена локального и удаленного блоков будут изменены после использования этих команд, то ключи аутентификации станут недостоверными и потребуется переконфигурировать клиентскую часть. Ключи MD5 и SHA основаны на паролях пользователей и идентификаторах блоков.

- Указание шаблона группового доступа для SNMP-пользователей (*необязательно*)

Система COS	`set snmp group [ -hex ] groupname user {-hex} username security-model {v1 | v2c | v3} [volatile | nonvolatile]`
Система IOS	`snmp-server group [groupname {v1 | v2c | v3 {auth|noauth}}] {read readview} [write writeview] [notify notifyview] {access acc-list}` (режим глобальной конфигурации)

Шаблон *groupname* определяет стратегии безопасности, которая будет использоваться группами SNMP-пользователей. Используемая группой версия протокола SNMP устанавливается с помощью ключевых слов v1, v2c и v3. Для протокола SNMP версии 3 (только для IOS-коммутаторов) уровень безопасности также должен быть указан с помощью параметров **auth** (пакетная аутентификация без шифрования), **noauth** (без пакетной аутентификации) или **priv** (пакетная аутентификация с шифрованием).

В IOS-коммутаторе можно также определять SNMP-снимки, ограничивающие MIB-доступ для группы, с помощью ключевых слов **read** (снимок *readview* определяет считываемые объекты; согласно стандартным установкам для всего Internet OID-пространства 1.3.6.1), **write** (снимок *writeview* определяет записываемые объекты; нет стандартного доступа для записи) и **notify** (снимок *notifyview* определяет извещения, которые могут быть отправлены группе; стандартных установок нет). Для дальнейшего ограничения SNMP-доступа группы можно использовать необязательный стандартный список IP-доступа (*acc-list*).

В COS-коммутаторах SNMP-пользователь должен быть определен как часть данной группы.

- Указание SNMP-пользователей и методов доступа (*необязательно; только для IOS-коммутаторов*).

В версиях 1 и 2 протокола SNMP для подключения пользователя к группе используется следующая команда.

Система COS	Нет
Система IOS	`snmp-server user username groupname [remote ip address] {v1 | v2c} [access acc-list]` (режим глобальной конфигурации)

Пользователь, определяемый с помощью параметра *username*, принадлежит групповому шаблону *groupname*. IP-адрес удаленного SNMP-диспетчера, которому принадлежит пользователь, может быть указан с помощью ключевого слова **remote**. Версию протокола SNMP необходимо указывать с помощью ключевых слов v1 и v2c. Применяя ключевое слово **access**, можно использовать стандартный список IP-доступа, разрешающий только определенные адреса отправителей для SNMP-пользователя.

- В третьей версии протокола SNMP пользователь подключается к группе и стратегиям безопасности путем ввода следующей команды:

Система COS	Нет
Система IOS	`snmp-server user` *username groupname* [`remote` *ip-address*] `v3` [`encrypted`] [`auth` {`md5` | `sha`} *auth-password*] [`access` *acc-list*] (режим глобальной конфигурации)

Пользователь с именем, заданным параметром *username*, определяется как принадлежащий групповому шаблону *groupname*. IP-адрес удаленного SNMP-диспетчера, которому принадлежит пользователь, может быть указан с помощью ключевого слова `remote`. Третья версия протокола SNMP должна быть указана с помощью ключевого слова `v3`. Применяя ключевое слово `access`, можно использовать стандартный список IP-доступа, разрешающий только определенные адреса отправителей для SNMP-пользователя.

Стандартно пароли для пользователей вводятся в виде текстовых строк. Если используется ключевое слово `encrypted`, то пароли необходимо вводить как MD5-дайджесты (уже зашифрованы). Пароль аутентификации для пользователя указывается с помощью ключевого слова `auth`. Тип аутентификации – с помощью ключевых слов `md5` (HMAC-MD5-96 Message Digest 5) и `sha` (HMAC-SHA-96) и текстовой строки *auth-password* (длиной до 64 символов).

в) Ограничение функций коммутатора, управляемых с помощью протокола SNMP (*необязательно; только для IOS-коммутаторов*).

- Включение использования операции SNMP-перезагрузки.

Система COS	Нет
Система IOS	`snmp-server system-shutdown` (режим глобальной конфигурации)

Стандартно пользователь не может использовать протокол SNMP для запуска операций перезагрузки в коммутаторе. Если эта функция желательна, можно использовать приведенную выше команду для включения управления перезагрузкой.

- Указание операций TFTP-сервера, управляемых с помощью протокола SNMP.

Система COS	Нет
Система IOS	`snmp-server tftp-server-list` *acc-list* (режим глобальной конфигурации)

Протокол SNMP можно использовать для того, чтобы заставить коммутатор сохранять или загружать свой конфигурационный файл на TFTP сервер. Можно использовать стандартный список IP-доступа (*acc-list*) для разрешения только ограниченного набора IP-адресов TFTP сервера.

3. Конфигурирование SNMP-извещений (*необязательно*).

а) Указание глобального списка отправляемых извещений.

Система COS	`set snmp trap {enable | disable} type`
Система IOS	`snmp-server enable {traps [type] [option] | informs}` (режим глобальной конфигурации)

Функция отправки извещений включена для указанных типов (как прерываний, так и информационных запросов). Поскольку команда позволяет задать только один тип, ее нужно вводить необходимое количество раз. В IOS-коммутаторе, если ключевое слово type не указано, включаются все доступные извещения. Кроме того, если команда не внесена по крайней мере один раз, то ни одно из контролируемых ею извещений не будет включено.

В IOS-коммутаторе возможными вариантами значений параметра type являются: `c2900` (извещение, относящееся ко Catalyst серии 2900), `cluster` (административные изменения кластера), `config` (конфигурационные изменения), `entity` (изменения MIB объекта), `hsrp` (изменения состояния HSRP), `vlan-membership` (изменения в составе VLAN-сети порта) и `vtp` (события протокола магистральных сетей VLAN). Для типа (type) `snmp` (извещения состояния основного маршрутизатора) можно использовать ключевое слово option с параметрами `authentication` (сбои аутентификации), `linkup` (включение интерфейса), `linkdown` (отключение интерфейса) и `coldstart` (повторная инициализация маршрутизатора). Если не используется ни одно из указанных ключевых слов, то все соответствующие функции включены.

В COS-коммутаторе возможными вариантами параметра type являются: `all` (включение всех типов прерываний), `auth` (сбои аутентификации), `bridge` (изменение корня и топологии STP), `chassis` (предупреждение шасси), `config` (конфигурационные изменения), `entity` (прерывания MIB-объекта), `entityfru` (прерывания заменяемого блока), `envfan` (сбои в работе вентилятора), `envpower` (события блока питания), `envshutdown` (выключение окружения), `ippermit` (отказы списков разрешения IP доступа), `module` (включение-отключение модуля коммутатора), `repeater` (RFC 1516 Ethernet события повторителя), `stpx` (STPX-прерывание), `syslog` (Syslog-извещения), `system`, `vmps` (изменение состава VLAN-сети) и `vtp` (события протокола VTP).

б) Указание получателей извещений

Система COS	SNMPv1 и SNMPv2c: `set snmp trap host` *community-string* SNMPv3: `set snmp targetaddr [-hex] host param [-hex] {paramname} {ipaddr} [udpport {port}] [timeout {value}] [retries {value}] [volatile | nonvolatile] [taglist {[-hex]tag}] [[-hex] tag tagvalue] set snmp targetparams [-hex] {paramname} user [-hex] {username} {security-model v3} {message-processing v3 {noauthentication | authentication | privacy}} [volatile | nonvolatile]`
Система IOS	`snmp-server host host [traps | informs] [version {1 | 2c | 3 [ auth | noauth]}]` *community-string* `[udp-port` *port*`]` *[type]* (режим глобальной конфигурации)

В качестве получателя SNMP-извещения (`trap` или `inform`) указывается один узел (параметр *host* — либо IP-адрес, либо имя узла). Версия протокола SNMP может быть задана с помощью необязательного ключевого слова `1` (SNMPv1, стандартное значение), `2c` (SNMPv2) или `3` (SNMPv3). Если используется третья версия протокола SNMP, можно задать ключевое слово для выбора типа безопасности: `auth` (использование MD5- и SHA-аутентификации) или `noauth` (без аутентификации или обеспечения частного доступа; стандартная установка).

Ключевое слово *community-string* указывает пароль, который совместно используется SNMP-агентом и диспетчером. Используемый UDP-порт может быть задан с помощью параметра *port* (по умолчанию — 162).

В IOS-коммутаторе возможными вариантами значений параметра *type* являются: `c2900` (извещения, основанные на Catalyst серии 2900), `cluster` (административные изменения кластера), `config` (конфигурационные изменения), `entity` (изменения MIB-объекта), `hsrp` (изменения состояния HSRP), `vlan-membership` (изменения в составе VLAN-сети порта) и `vtp` (события протокола магистральных сетей VLAN). Для типа (*type*) `envmon` (изменения состояния основного коммутатора) можно использовать ключевое слово *option* с параметрами `authentication` (сбои аутентификации), `linkup` (включение интерфейса), `linkdown` (отключение интерфейса) и `coldstart` (повторная инициализация коммутатора). Если не используется ни одно из указанных ключевых слов, то все соответствующие функции включены.

в) **Настройка параметров извещения** (*необязательно; только для IOS-коммутаторов*).

- Указание параметров прерываний.

Система COS	Нет
Система IOS	`snmp-server trap-timeout` *seconds* `snmp-server queue-length` *length* (обе команды вводятся в режиме глобальной конфигурации)

Отправка SNMP-прерываний не является надежной, поскольку подтверждения не запрашиваются. Прерывания могут устанавливаться в очередь и отправляться повторно, только когда отсутствует маршрут к получателю прерываний. В таком случае маршрутизатор ожидает в течение времени, заданного параметром *seconds* (стандартно — 30 секунд), прежде чем повторно отправить прерывание. Кроме того, стандартно в очередь может устанавливаться десять прерываний для каждого получателя. Можно использовать команду `queue-length` для установки размера очереди, равного значению параметра *length* каждого прерывания.

- Указание адреса отправителя для извещений

Система COS	Нет
Система IOS	`snmp-server trap-source` *interface* (режим глобальной конфигурации)

Прерывания протокола SNMP могут отправляться с любого доступного интерфейса коммутатора. Чтобы коммутатор отправлял все прерывания, не-

пользуя единственный IP-адрес отправителя, необходимо указать используемый интерфейс (параметр *interface*). Этот способ позволяет упростить сопоставление прерываний с коммутатором отправителя.

г) Включение прерываний SNMP-канала на определенных интерфейсах.

Система COS	`set port trap` *mod/port* `{enable | disable}`
Система IOS	`[no] snmp trap link-status` (режим конфигурирования интерфейса)

IOS-коммутаторы стандартно генерируют прерывания SNMP-канала на всех интерфейсах, когда они включаются или отключаются. Если такие прерывания нежелательны, следует использовать ключевое слово `no` для отключения прерываний на определенных интерфейсах. Стандартной установкой для COS-коммутаторов является ключевое слово `disable`, отключающее прерывание на всех портах.

4. Включение RMON-поддержки (*необязательно*):

а) Накопление RMON-статистики (*необязательно*):

Система COS	`set snmp rmon {enable | disable}`
Система IOS	`rmon collection stats` *index* `[owner` *name*`]`

В COS-коммутаторе RMON-статистика накапливается для всех Ethernet-, Fast Ethernet-, Gigabit Ethernet- и EtherChannel-портов. В то же время в IOS-коммутаторах RMON-статистика накапливается только на сконфигурированных интерфейсах. Статистические данные собираются в *семейства* (collections), каждое из которых уникально и определяется номером семейства или индексом (параметр *index*, от 1 до 65535). Необязательный параметр *name* (имя владельца – текстовая строка) может быть задан для связи имени пользователя с определенным семейством.

б) Сбор статистических данных RMON-истории (*необязательно, только для IOS-коммутаторов*):

Система COS	Нет

Система IOS	`rmon collection history` *index* `[owner` *name*`]`
	`[buckets` *nbuckets*`]` `[interval` *seconds*`]`
	(режим конфигурирования интерфейса)

IOS-коммутатор способен накапливать историческую статистику на сконфигурированных интерфейсах. Статистические данные собираются в семейства (collections), каждое из которых уникально и определяется номером семейства или индексом (параметр *index*, от 1 до 65535). Необязательный параметр *name* (имя владельца – текстовая строка) может быть задан для связи имени пользователя с определенным семейством. Ключевое слово `buckets` определяет количество предназначенных для использования ячеек семейств (стандартно 50). Ключевое слово `interval` определяет длительность сеанса сбора в секундах (стандартно – 1800 секунд).

в) Конфигурирование RMON-предупреждения *(необязательно; только для IOS-коммутаторов)*.

Система COS	Нет
Система IOS	`rmon alarm` *number object interval* {**delta** | **absolute**} **rising-threshold** *rise* [*event*] **falling-threshold** *fall* [*event*] [**owner** *string*] (режим глобальной конфигурации)

Отмеченное ключевым словом *number* от 1 до 65535 предупреждение конфигурируется для мониторинга определенной MIB-переменной (*object*). Объект задается в виде десятичного значения, разделенного точками, *entry.integer.instance*. Поле *interval* определяет период времени в секундах (от одной до 4294967295), в течение которого предупреждение будет отслеживать объект. Ключевое слово **delta** позволяет отслеживать относительные изменения MIB-переменных, тогда как ключевое слово **absolute** отслеживает MIB-переменные непосредственно. Можно настроить предупреждение таким образом, чтобы объект тестировался в сравнении с параметрами **rising-threshold** и **falling-threshold**, где поля *rise* и *fall* являются пороговыми значениями, которые вызывают данное предупреждение. Поле *event* определяет номер события в таблице событий для выдачи порогов восстановления и отключения. Необязательный параметр *owner* (*текстовая строка*) может быть задан для определения владельца предупреждения.

г) Конфигурирование RMON-события *(необязательно, только для IOS-коммутаторов)*.

Система COS	Нет
Система IOS	`rmon event` *number* [**description** *string*] [**owner** *name*] [**trap** *community*] [**log**] (режим глобальной конфигурации)

RMON-события идентифицируются по произвольному номеру (параметр *number*, от 1 до 65535). Ключевое слово **description** задает описательную строку события (*string* – текстовая строка). Необязательному параметру **owner** (владелец события) может быть присвоено имя (*name* – текстовая строка). Если задано ключевое слово **trap**, то SNMP-прерывание генерируется со строкой сообщества (*community* – текстовая строка). Использование ключевого слова **log** приводит к тому, что событие генерирует запись в RMON-журнале на коммутаторе.

Пример конфигурирования протокола SNMP

В этом примере коммутатор конфигурируется для использования протокола SNMP с применением сообщества *public* для доступа с правами лишь чтения или сообщества *root-secret* для доступа с правами чтения записи. SNMP-доступ только для чтения предоставляется любому узлу в сети 172.30.0.0 и узлам управления сетью 172.30.5.91 и 172.30.5.95 для доступа с правами на чтение и запись. (Такие настройки можно реа-

лизовать с помощью списков доступа на IOS-коммутаторе. Однако COS-коммутатор ограничен командами списков разрешения IP-доступа для всех типов SNMP-доступа. В список разрешения IP-доступа необходимо добавить определенные узлы.)

SNMP-прерывания отправляются на машину SNMP-агент с адресом 172.30.5.93 со строкой сообщества nms. Отправляются все возможные прерывания, кроме прерываний изменения конфигурации коммутатора. Также для порта 3/1 отключены прерывания включения/отключения SNMP-канала.

Система COS	`set system contact John Doe, Network Operations`
	`set system location Building A, closet 123`
	`set snmp community read-only public`
	`set snmp community read-write noc-team`
	`set snmp trap 172.30.5.93 nms`
	`set snmp trap enable all`
	`set snmp trap disable config`
	`set ip permit 172.30.5.91`
	
	`set ip permit 172.30.5.95`
	`set ip permit enable snmp`
	`set port trap 3/1 disable`

Система IOS	`snmp-server contact John Doe, Network Operations`
	`snmp-server location Building A, closet 123`
	`snmp-server community public ro 5`
	`snmp-server community noc-team rw 6`
	`snmp-server host 172.30.5.93 traps nms`
	`snmp-server enable traps`
	`no snmp-server enable config`
	`access-list 5 permit 172.30.0.0 0.0.255.255`
	`access-list 6 permit host 172.30.5.91`
	`access-list 6 permit host 172.30.5.95`
	`interface gig 3/1`
	(все команды, указанные выше, вводятся в режиме глобальной конфигурации)
	`no snmp trap link-status`
	(режим конфигурирования интерфейса)

Отображение информации о настройках протокола SNMP

В табл. 12.3 перечислены некоторые команды коммутатора, которые можно использовать для отображения полезной информации о протоколе SNMP.

Таблица 12.3. Команды коммутатора для отображения SNMP-информации

Функция отображения	Операционная система коммутатора	Команда
SNMP-конфигурация	COS	`show snmp`
	IOS	`show snmp`

Функция отображения	Операционная система коммутатора	Команда
RMON-семейства	COS	нет
	IOS	`show rmon [alarms | events | history | statistics]`

12.3: анализатор коммутируемых портов

- Анализатор коммутируемых портов (*Switched Port Analyzer — SPAN*) отражает в коммутаторе трафик от одного или нескольких портов отправителя или VLAN-сети отправителя на порт назначения. Это позволяет подключить к порту назначения устройство мониторинга, такое, как сетевой анализатор, для перехвата трафика
- Порты SPAN-отправителя и получателя должны располагаться на одном физическом коммутаторе.
- Для обеспечения одновременного мониторинга можно сконфигурировать множество SPAN-сеансов.
- Удаленный SPAN-анализатор (*Remote SPAN — RSPAN*) обеспечивает мониторинг трафика от отправителя на одном коммутаторе к получателю на одном или нескольких удаленных коммутаторах.
- RSPAN-трафик транспортируется от отправителя к получателю через специальную сеть RSPAN VLAN.
- Функция RSPAN доступна только в коммутаторах семейства Catalyst 4000 и 6000.

Внимание!

Рассмотрим ситуацию, когда скорости между портами SPAN-отправителя и получателя не совпадают. Во время SPAN-сеанса коммутатор просто копирует пакеты от отправителя и помещает их в исходящую очередь порта получателя. Если на порту назначения возникает перегрузка, то SPAN-пакеты будут удаляться из очереди и не будут видны у получателя. Таким образом, какие-либо перегрузки на SPAN-получателе не влияют на трафик от SPAN-отправителя.

Конфигурирование SPAN-анализатора

1. Создание SPAN-сеанса.

   Выбор отправителя и получателя *только для COS- и IOS-коммутаторов Catalyst 2900/3500*.

Система COS	`set span` *src-mod/src-ports / src-vlans* `|both|` *dest-mod/dest-port* `[rx | tx |both] [inpkts {enable | disable}] [learning {enable | disable}] [multicast {enable | disable}] [filter vlans...] [create]`
Система IOS	`interface` *dest-interface* (режим глобальной конфигурации) `port monitor {`*src-interface* `| vlan` *src-vlan*`}` (режим конфигурирования интерфейса)

Отправителем трафика для SPAN-сеанса могут быть порты коммутатора, VLAN-сеть или административный интерфейс коммутатора (только для COS-коммутаторов). Если требуется обеспечить мониторинг портов коммутатора, то они идентифицируются параметрами *src-mod/src-ports* (COS; может указываться один порт или диапазон портов) или параметрами *src-interface* IOS: только один тип и номер интерфейса).

Если осуществляется мониторинг VLAN-сети, то они идентифицируются параметрами *src vlans* (система COS допускает одну VLAN-сеть или диапазон VLAN-номеров, а IOS разрешается использовать только один номер VLAN-сети). COS-коммутаторы при необходимости также допускают использование в качестве отправителя административного интерфейса sc0.

Порт SPAN-получателя, к которому подключено мониторинговое устройство, идентифицируется с помощью параметров *dest-mod/dest-port* (COS). Для IOS-коммутаторов прежде чем ввести команду `port monitor`, необходимо выбрать интерфейс получателя с помощью команды `interface` *dest-interface*. В IOS-коммутаторах порт получателя должен принадлежать той же VLAN-сети, что и отправитель.

Кроме того, можно выбрать направление исследуемого трафика отправителя с помощью ключевых слов `rx` (трафик, получаемый отправителем), `tx` (трафик, отправляемый отправителем) и `both` (в обоих направлениях; стандартная установка) для IOS-коммутаторов характерен мониторинг трафика в обоих направлениях.

Стандартно многоадресный трафик изучается по мере поступления от отправителя. Для отключения этого режима используются ключевые слова `multicast enable`.

Порт получателя обычно используется для перехвата трафика, поэтому по умолчанию на порту получателя не допускается появление входящего трафика. Если это необходимо, можно включить обычную коммутацию входящего трафика на устройстве получателя с помощью ключевых слов `inpkts enable`.

Внимание!

Следует помнить о том, что порт получателя всегда принадлежит активной VLAN-сети независимо от того, осуществляет он мониторинг для SPAN-сеанса или нет. Кроме того, для того чтобы можно было осуществлять мониторинг BPDU-блоков протокола STP, в порту SPAN-получателя этот протокол не используется. Следовательно, если входящие пакеты разрешены (`inpkts`) и порт получателя подключен к другому сетевому устройству, то возможно формирование петель распределенного связующего дерева.

Стандартно в порту получателя изучаются MAC-адреса входящих пакетов, появляющихся на любом порту коммутатора. Можно отключить изучение адресов на получателе с помощью ключевых слов `learning disable`.

Если в качестве порта отправителя используется магистральный порт, то в целях мониторинга можно отфильтровать определенные VLAN-сети. В COS-коммутаторе для этого используются ключевые слова `filter vlans` (одна сеть или диапазон VLAN-номеров). В IOS-коммутаторе подобное средство не предусмотрено. Однако можно назначить порт получателя той сети VLAN, мониторинг которой необходимо осуществить в магистральном канале отправителя.

Совет

Если осуществляется мониторинг отправителей, которые принадлежат нескольким VLAN-сетям, то может понадобиться делать записи об исходных VLAN-сетях для прибывающих в порт получателя пакетов. Для этого необходимо включить функцию магистрализации передачи данных на порту получателя. Пакеты отправителя будут маркироваться VLAN-номерами тех сетей, из которых они поступали.

Совет

COS-коммутаторы допускают существование нескольких активных SPAN-сеансов. Первый сеанс можно конфигурировать так, как показано в полном синтаксисе для COS-команды `set span`. Для создания последующих сеансов используется ключевое слово `create`. Если оно опущено, то вновь сконфигурированный сеанс перезаписывает первый сеанс.

2. Выбор отправителя и получателя (*IOS; только для коммутатора Catalyst 6000*).

   а) Выбор отправителя сеанса.

Система COS	Нет
Система IOS	`monitor session` *session* `{source {interface` *interface*`}` `| {vlan` *vlan-id*`}} [,	-	rx	tx	both]` (режим глобальной конфигурации)

   SPAN-сеанс уникально идентифицируется с помощью параметра *session* (1 или 2). В качестве отправителя может использоваться *interface* (параметр интерфейс — тип и номер интерфейса или номер порта канала) или номер VLAN-сети (*vlan-id*, от 1 до 1005). Множество VLAN-сетей отправителей можно задать с помощью ключевого слова `vlan`, за которым следуют номера (*vlan-id*), разделенные запятыми (,). Для указания диапазона VLAN-номеров используется ключевое слово `vlan`, за которым следует первый и последний номера (*vlan-id*), разделенные дефисом (-).

   Можно осуществлять мониторинг трафика получателя в различных направлениях. Для этого используется одно из ключевых слов: `rx` (трафик, получаемый отправителем), `tx` (трафик, отправляемый отправителем) или `both` (оба направления; стандартная установка).

   б) Выбор получателя сеанса.

Система COS	Нет
Система IOS	`monitor session {destination {interface` *interface*`} [,	-]	{vlan` *vlan-id*`}}` (режим глобальной конфигурации)

   В качестве получателя для SPAN-сеанса (номер сеанса 1 или 2) может использоваться интерфейс (параметр *interface* — тип и номер интерфейса) или номер VLAN-сети (параметр *vlan-id*, от 1 до 1005). Также при необходимости можно указать множество получателей. Задать множество можно с помощью ключевого слова `interface`, за которым следует список номеров интерфейса (параметр *interface*), разделенных запятыми (,). Для определения диапазона интерфейсов используется ключевое слово `interface`, за которым следует пер-

ный и последний номера интерфейса (параметр `interface`), разделенные дефисом (-).

в) Фильтрация VLAN-сетей на магистральном канале отправителя (*необязательно*).

Система COS	Нет
Система IOS	`monitor session` *session* `filter vlan` *vlan-id*} [, | -] (режим глобальной конфигурации)

Если в качестве порта отправителя используется магистральный канал, то его можно фильтровать для выбора определенных VLAN-сетей. VLAN-номер указывается с помощью параметра *vlan-id* (от 1 до 1005). Можно задать множество VLAN-сетей отправителей с помощью ключевого слова `vlan`, за которым следует список номеров сетей (*vlan-id*), разделенных запятыми (,). Для определения диапазона VLAN-сетей используется ключевое слово `vlan`, за которым следует первый и последний номера сетей (*vlan-id*), разделенные дефисом (-).

3. Отключение SPAN-сеанса (*необязательно*).

Система COS	`set span disable` [*dest-mod/dest-port* | `all`]
Система IOS	`no monitor session` *session* (режим глобальной конфигурации) или `no port monitor` (режим конфигурирования интерфейса)

SPAN-сеансы могут отключаться индивидуально, при этом ссылка на них осуществляется с помощью параметра *dest-port* (COS) или номера сеанса *session* (IOS). В IOS коммутаторах Catalyst 2900/3500 SPAN сеансы отключаются на интерфейсах получателя с помощью команды `no port monitor`.

Конфигурирование службы RSPAN

1. Создание одной или нескольких VLAN-сетей, используемых в RSPAN-мониторинге

Система COS	`set vlan` *vlan-id* `rspan`
Система IOS	Нет

VLAN-сеть номер *vlan-id* (от 1 до 1000, 1025 до 4024) должна быть создана на всех коммутаторах от RSPAN-отправителя до RSPAN-получателя. Также в RSPAN VLAN-сети следует обеспечить сквозной магистральный режим, поскольку эта сеть транспортирует трафик, для которого осуществляется удаленный мониторинг. Для каждого RSPAN-сеанса, который будет использоваться, необходимо создать отдельную RSPAN VLAN-сеть. Более подробная информация, касающаяся конфигурации VLAN-сетей и протокола VTP, приведена в главе 6, "VLAN-сети и транкинг".

Внимание!

Необходимо отметить использование ключевого слова `rspan` при создании SPAN VLAN-сети. Оно необходимо для корректной транспортировки RSPAN-трафика VLAN-сетью.

Коммутатор, поддерживающий RSPAN, при попытке отправить RSPAN-пакеты получателю лавинно отправляет их через все свои порты, принадлежащие данной RSPAN-сети. Это происходит вследствие того, что коммутатор, участвующий в RSPAN-мониторинге, не имеет сведений о том, где расположен получатель.

В противном случае, если бы коммутатор использовал обычную VLAN-сеть, он пытался бы перенаправить RSPAN-пакеты на порты, где были обнаружены адреса получателей, что в целом совершенно отличается от RSPAN-мониторинга. Именно поэтому все коммутаторы, задействованные в сквозном RSPAN-мониторинге, должны быть RSPAN-совместимыми. В настоящее время круг таких коммутаторов ограничен моделями Catalyst 4000 и 6000.

Совет

Рекомендуется специально в целях мониторинга создать и поддерживать RSPAN VLAN-сеть. Не следует разрешать любым обычным узлам подключаться к этой сети.

В идеальном случае все коммутаторы будут принадлежать общему VTP-домену, поэтому VLAN-сеть может быть создана на VTP-сервере и распространена среди всех остальных коммутаторов. Функция VTP-отсекания также отсекает RSPAN VLAN от излишних магистральных каналов, ограничивая влияние трафика в неиспользуемых областях сети.

Следует помнить о том, что RSPAN-трафик способен увеличивать нагрузку на магистральный канал, даже если RSPAN-мониторинг ограничен только одной специальной VLAN-сетью в этом магистральном канале. Если дополнительная нагрузка значительна, то обычный и изучаемый трафики конкурируют друг с другом, в результате чего возможны потери данных в обеих.

---

2. Выбор исследуемых отправителей (*только для коммутаторов-отправителей*)

Система COS	`set rspan source` *{src-mod/src-ports... | vlans... | sc0} {rspan-vlan} [rx | tx | both] [multicast {enable | disable}] [filter vlans...] [create]*
Система IOS	Нет

RSPAN-отправитель идентифицируется как один или несколько физических портов коммутаторов (*src-mod/src-ports*), как одна или несколько VLAN-сетей с номерами *vlans* или как административный порт sc0. Это выполняется только на коммутаторе, к которому подключен отправитель. Номер, используемый для RSPAN VLAN сети, задается с помощью параметра *rspan vlan* (от 1 до 1000, от 1025 до 4094). Направление исследуемого трафика может быть задано с помощью ключевых слов `rx` (трафик, получаемый отправителем), `tx` (трафик, передаваемый отправителем) или `both` (оба направления; стандартная настройка). Стандартно многоадресатный трафик изучается по мере его выхода от отправителя. Для отключения этого режима используется ключевая фраза `multicast disable`.

Если в качестве порта отправителя используется магистральный канал, то можно фильтровать его для выбора определенных VLAN-сетей с целью их мониторинга, используя ключевые слова `filter vlans` (одна VLAN-сеть или диапазон VLAN-номеров).

Совет

На коммутаторе отправителя можно сконфигурировать несколько активных RSPAN-сеансов. Первый сеанс создается согласно приведенным выше рекомендациям. Для создания последующих сеансов используется ключевое слово `create`. Если оно опущено, то вновь сконфигурированный сеанс перезаписывает первый сеанс. Для каждого сеанса следует использовать отдельную RSPAN VLAN-сеть.

3. Выбор получателей (*только коммутаторы получателей*).

Система COS	`set rspan destination` *mod/port* `[rspan-vlan] [inpkts {enable | disable}] [learning {enable | disable}] [create]`
Система IOS	Нет

Порт SPAN-получателя, к которому подключено мониторинговое устройство, идентифицируется параметром *mod/port*. Это выполняется только на том коммутаторе, на котором расположен порт получателя.

Порт получателя обычно используется для устройства перехвата трафика, поэтому появление входящего трафика на порту получателя не допускается по умолчанию. При необходимости можно включить обычную коммутацию входящего трафика на порту получателя с помощью ключевых слов `inpkts enable`.

Внимание!

Мониторинг RSPAN отличается от SPAN тем, что на порту получателя всегда включен протокол STP, предотвращающий случайное формирование мостовых петель в случае, если другие сетевые устройства подключаются к порту получателя. Однако это также означает, что с помощью RSPAN-функции невозможно осуществлять мониторинг STP BPDU-пакетов.

Стандартно включена функция определения MAC-адресов из входящих пакетов, поступающих на какой-либо порт коммутатора. Можно отключить эту функцию на порту получателя с помощью ключевой фразы `learning disable`.

Совет

На коммутаторе получателя также можно сконфигурировать несколько активных RSPAN-сеансов. Первый сеанс создается согласно приведенным выше рекомендациям. Для создания последующих сеансов используется ключевое слово `create`. Если оно опущено, то вновь сконфигурированный сеанс перезаписывает первый сеанс. Для каждого сеанса следует использовать отдельную RSPAN VLAN-сеть.

4. Дальнейшая конфигурация не требуется (*только для промежуточных коммутаторов*).

Коммутаторы на маршруте от RSPAN-отправителя к получателю не нуждаются в информации о какой-либо специфической RSPAN-конфигурации. В конечном итоге RSPAN VLAN-сети создаются в сквозном режиме, и промежуточные коммутаторы лишь распространяют RSPAN-трафик в верном направлении к получателям. Необходимо помнить, что все промежуточные коммутаторы должны поддерживать функцию RSPAN.

5. Отключение RSPAN-сеанса (*необязательно*).

Система COS	`set rspan disable source [rspan-vlan | all]`
	или
	`set rspan disable destination [mod/port | all]`

Система IOS	Нет

RSPAN-сеанс, потребность в котором отпала, можно отключить. Сеансы отправителей идентифицируются параметром *rspan-vlan* (номер VLAN сети) или ключевым словом `all`. Сеансы получателей идентифицируются параметром *mod/port* (порт получателя) или ключевым словом `all`.

Примеры конфигурирования SPAN-анализаторов

Сетевой анализатор А (перехватчик пакетов — sniffer) подключен к порту коммутатора Catalyst 5/1 и настраивается на мониторинг всего трафика в сети VLAN 58.

К коммутатору Catalyst подключен персональный компьютер (порт 4/39), а также файловый сервер (порт 2/4). Сетевой анализатор Б подключен к порту 5/48. В коммутаторе настраивается SPAN-сеанс, который позволит анализатору перехватывать весь трафик, направленный к серверу и от него. На рис. 12.1 представлена диаграмма сети для двух SPAN-сеансов.

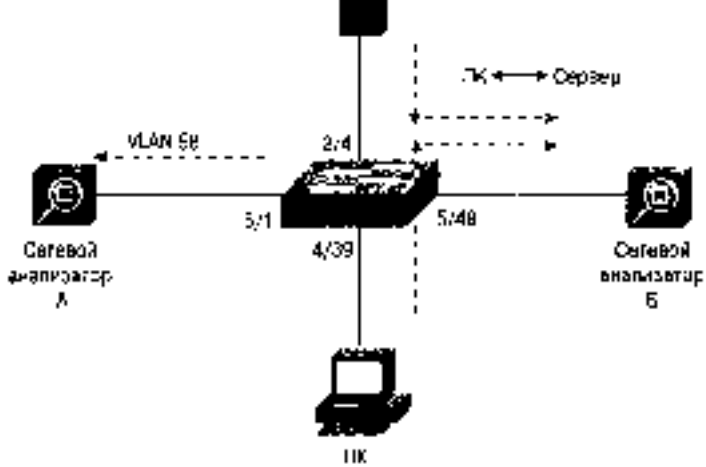


Рис. 12.1. Сетевая диаграмма для примера SPAN-конфигурации

Система COS	`set span 58 5/1 both` `set span 2/4 5/48 both create`

Система IOS	`interface fast 5/1`
	`port monitor vlan 58`
	`interface fast 5/48`
	`port monitor fast 2/4`

IOS Catalyst 6000	`monitor session 1 source vlan 50 both`
	`monitor session 1 destination interface fast 5/1`
	`monitor session 2 source interface fast 5/48 both`
	`monitor session 2 destination interface fast 2/4`
	(режим глобальной конфигурации)

На рис. 12.2 приведена сеть, состоящей из трех коммутаторов. Файловый сервер подключен к порту 3/1 коммутатора Catalyst B. Сетевой анализатор подключен к порту 5/48 коммутатора Catalyst C. Коммутатор Catalyst A подключен к коммутаторам Catalyst B и C с помощью двух магистральных портов. Весь RSPAN-трафик от отправителя к получателю транспортируется по RSPAN VLAN-сети 901. (Предположим, что в данном случае коммутатор Catalyst B является VTP-сервером для доклада из трех коммутаторов.)

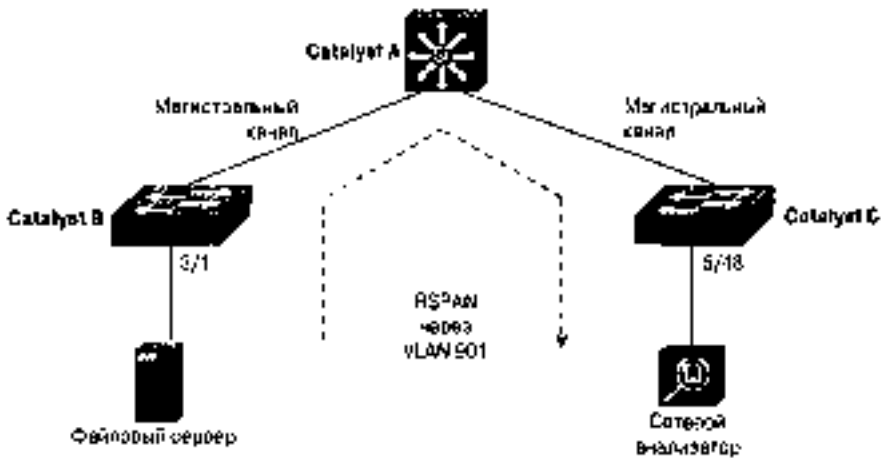


Рис. 12.2. Сетевая топология для примера RSPAN-конфигурации

Ниже приводится конфигурация коммутатора Catalyst B.

Система COS	`set vlan 901 rspan` `set rspan source 3/1 901 both`
Система IOS	Нет

Ниже приведена конфигурация коммутатора Catalyst C.

Система COS	`set rspan destination 5/48 901`
Система IOS	Нет

Никакой дополнительной конфигурации для коммутатора Catalyst A не требуется, поскольку он только транспортирует RSPAN-трафик по VLAN-сети 901.

Отображение информации о SPAN-конфигурации

В табл. 12.4 приведены команды коммутатора, которые можно использовать для отображения полезной информации о SPAN-конфигурации.

COS-коммутаторы отображают порты отправителей с метками "Admin Source" (административный отправитель) и "Oper Source" (операционный отправитель). Все

VLAN-сети или порты, которые настроены как отправители, перечисляются как административные отправители. В качестве операционных отправителей перечисляются только те порты, для которых осуществляется активный мониторинг (не отключены).

Таблица 12.4. Команды коммутатора для отображения SPAN-информации

Функция отображения	Операционная система коммутатора	Команда
Активность SPAN-сеанса	COS	`show span`

	IOS	`show monitor [session` *session-number*`]`
		или
		`show port monitor`

Активность RSPAN-сеанса	COS	`show rspan`
	IOS	Нет

12.4: управление питанием

- Функция управления питанием доступна только в коммутаторах семейства Catalyst 6000.
- Источники питания могут быть переведены в режим избыточности, при котором несколько источников питания разделяют общую нагрузку. Если один из них выходит из строя, то другой поддерживает полную системную нагрузку.
- Мощности источников питания, работающих без избыточности, могут объединяться для поддержки системы. Это полезно в ситуациях, когда общая нагрузка модулей коммутатора превышает мощность одного источника питания. Если один источник питания выходит из строя, а другой не обладает достаточной мощностью для поддержки нагрузки всей системы, то в целях сокращения нагрузки некоторые модули коммутатора отключаются.

Конфигурирование функции

1. Конфигурирование резервирования источников питания (*задача для коммутаторов Catalyst 6000*).

Система COS	`set power redundancy {enable disable}`
Система IOS	`power redundancy-mode {combined redundant}` (режим глобальной конфигурации)

Резервирование источников питания стандартно включено (`enable` или `redundant`).

2. Управление функциями источника питания (*только для IOS-коммутаторов Catalyst 6000*).

Система COS	Нет
Система IOS	`[no] power enable power-supply number`

Стандартно все источники питания включены. Источники питания идентифицируются по номеру (number — 1 или 2). Для отключения источника используется ключевое слово no

3. Управление питанием модулей коммутатора (*только для коммутаторов Catalyst 6000*.

Система COS	`set module power up | down mod`
Система IOS	`[no] power enable module mod` (режим глобальной конфигурации)

Стандартно все модули коммутатора получают питание. Для отключения питания какого-либо модуля используются ключевые слова down (COS) и no (IOS) с указанием номера модуля (mod, от 1 до максимального количества гнезд в шасси).

Отображение информации о функции управления питанием

В табл. 12.5 перечислены команды коммутатора, которые можно использовать для отображения полезных сведений о функции управления питанием.

Таблица 12.5. Команды коммутатора для отображения информации о функции управления питанием

Функция отображения	Операционная система коммутатора	Команда
Питание системы	COS	`show environment power`
	IOS	`show power [{available | redundancy-mode | {power-supply` *number*`} | total | used}]`
Состояние питания модулей	COS	`show module`
	IOS	`show power status all`

12.5: мониторинг температуры

Внимание!

Команды мониторинга температуры в настоящее время доступны только в коммутаторах семейств Catalyst 6000 и 4000.

В табл. 12.6 перечислены команды коммутатора, которые служат для отображения полезных сведений о мониторинге температуры.

В информации, отображаемой с помощью команды `show environment temperature`, перечислены результаты измерений температуры в момент включения и при перегрузке каждого модуля. В скобках приводится соответствующее предупреждение и указываются значения температуры, при достижении которых выдается критическое предупреждение. Для нормального функционирования температура коммутатора не должна превышать указанных в скобках уровней.

Таблица 12.6. Команды коммутатора для отображения информации о температуре устройств

Функция отображения	Операционная система коммутатора	Команда
Температурные показатели модулей	COS	`show environment temperature`
	IOS	`show environment temperature`

Значения в полях "Device 1" и "Device 2" относятся к дополнительным датчикам внутри модулей. Модули "VTT" расположены на объединительной плате шасси.

12.6: трассировка пакетов

- Команда `ping` (Packet Internet Groper — пакетное тестирование Internet) может применяться для тестирования сквозной связи от коммутатора к удаленному узлу. В IP `ping` используются ICMP-запросы, тип 8, и ICMP-ответы, тип 0.
- Команда `traceroute`, или трассировка маршрута третьего уровня, применяется для обнаружения маршрутизаторов вдоль маршрута, по которому пакеты направляются получателю. В IP-трассировке используются пакеты UDP-пробы на порт 33434.
- Команда `l2trace`, или трассировка маршрута второго уровня, используется для вычисления физического маршрута, по которому пакет следует через коммутируемую сеть.
- Команда `l2trace` осуществляет поиск получателя в таблице коммутации, а затем связывается со следующим соседним коммутатором посредством протокола CDP. В подобном режиме определяются все переходы между коммутаторами.
- Команда `l2trace` поддерживается лишь в коммутаторах Catalyst 4000, 5000 и 6000 (только COS-коммутаторы). Если вдоль маршрута к получателю встречаются другие коммутаторы, неспособные ответить на `l2trace`-запрос, то время ожидания команды будет увеличено.

Использование средств трассировки

1. Использование `ping`-пакетов для проверки достижимости устройства.

Система COS	`ping -s` *host [packet-size] [packet-count]*
Система IOS	`ping` *[host]* (режим непривилегированного пользователя)

Команда `ping` отправляет ICMP-пакеты, тип 8 (эхо-запрос), целевому узлу (*host* — IP-адрес или имя узла); в ответ ожидается получение ICMP эхо-ответов. COS-коммутатор отправляет один `ping`-пакет, если не используется параметр `-s`, который приводит к непрерывной отправке `ping`-пакетов, до тех пор пока выполнение команды не будет прервано нажатием клавиш <^>+<C> (<Ctrl>+<C>). Также можно указать размер `ping`-пакетов, *packet-size* (в байтах), и количество пакетов, *packet-count*.

Стандартно IOS-коммутатор отправляет получателю пять `ping` пакетов. Каждый пакет отображается с помощью одного из следующих символов:

- `!` — ответ успешно получен;
- `.` — отсутствие ответа в стандартном временном интервале — 2 секунды);
- `U` — получено сообщение об ошибке "получатель недостижим";
- `M` — получено сообщение о невозможности фрагментации;
- `C` — получен пакет индикации перегрузки;
- `I` — `ping`-тестирование прервано на коммутаторе;
- `?` — получен пакет неизвестного типа;
- `&` — время жизни пакета (Time-To-Live — TTL) истекло.

По завершении тестирования выдается отчет о скорости передачи, а также сводная информация о сквозной задержке: минимальное, среднее и максимальное значения в миллисекундах.

Внимание!

Для обычной команды `ping` может быть задан только адрес получателя. В качестве адреса отправителя в `ping`-пакетах используется адрес административного интерфейса коммутатора.

В IOS-коммутаторах также имеется более гибкая функция эхо-тестирования, которая называется *расширенным ping-тестированием (extended ping)*. Для его выполнения вводится команда `ping` в режиме привилегированного пользователя без параметров, после чего коммутатор запрашивает у пользователя все доступные `ping`-параметры, включая используемый адрес отправителя. Возможные параметры приведены ниже.

- Протокол (стандартно — `ip`). Также в IOS-коммутаторе Catalyst 6000 и других коммутаторах третьего уровня могут использоваться ключевые слова `appletalk`, `clns`, `novell`, `apollo`, `vines`, `decnet` и `xns`.
- Тестируемый адрес.
- Количество повторов (стандартно — 5 пакетов) — количество отправляемых эхо-пакетов.
- Размер дейтаграммы (стандартно 100 байтов) — размер эхо пакета. Для тестирования фрагментации пакетов следует выбрать размер, превышающий размер MTU (*Maximum Transfer Unit — максимально возможный блок передачи данных*).
- Время ожидания (стандартно — 2 секунды) — период времени ожидания ответа на каждый пакет-запрос.
- Расширенные команды.
  - Адрес или интерфейс отправителя — может быть задан любой адрес отправителя, однако если требуется получать пакеты-ответы, необходимо использовать адрес административного интерфейса данного коммутатора.
  - Тип обслуживания (стандартно — 0).

- Установка DF-бита[1] в IP-заголовок (стандартно не производится, но) если этот бит установлен, пакет не фрагментируется на маршруте с меньшим размером MTU. Эту функцию можно использовать для обнаружения наименьшего размера MTU в маршруте.
- Проверка данных ответов (по умолчанию не производится, но) — данные, отправляемые в пакете эхо-запроса, сравниваются с данными, полученными в эхо-ответе.
- Шаблон данных (стандартно — 0xABCD) — шестнадцатибитовое поле, которое повторяется в части данных пакета. Шаблон данных (data pattern) может оказаться полезным при тестировании целостности данных совместно с устройством CSU/DSU и для проверки кабельной системы.
- Параметры `loose`, `strict`, `record`, `timestamp` и `verbose` (по умолчанию не используются). `loose` (свободный маршрут от отправителя с адресами переходов), `strict` (строгий маршрут от отправителя с адресами переходов), `record` (запись маршрута с определенным количеством переходов), `timestamp` (запись временных меток на каждом транзитном маршрутизаторе) и `verbose` (включает режим вывода подробных сведений). Параметр `record` полезен при просмотре записей адресов маршрутизаторов, которые проходит пакет на сквозном маршруте.
- Отклонение размеров (стандартно не используется) — отправляет эхо-запросы с различными размерами пакетов:
  - минимальный размер (стандартно — 36);
  - максимальный размер (стандартно — 18024);
  - интервал (стандартно — 1).

2. Использование трассировки маршрута третьего уровня для обнаружения маршрутизаторов (*необязательно*).

Система COS	`traceroute [-n] [-w wait-time] [-i initial-ttl] [-m max ttl] [-p dest port] [-q nqueries] [-t tos] host [data-size]`
Система IOS	`traceroute [protocol] [host]` (режим непривилегированного пользователя)

Команда `traceroute` отправляет последовательные пробные пакеты узлу `host` (либо сетевой адрес, либо имя узла). В поле *protocol* в IOS-коммутаторе Catalyst 6000 и других коммутаторах третьего уровня могут быть указаны ключевые слова `appletalk`, `clns`, `ip` и `vines`.

При IP-трассировке первая группа пакетов (стандартно — 3) отправляется с TTL, равным 1. Первый маршрутизатор на маршруте уменьшает TTL-значение на единицу, определяет, что оно равно нулю, и возвращает ICMP-пакеты с сообщением об ошибке, связанном с превышением TTL. После этого отправляются последовательные наборы пакетов, в каждом из которых TTL-значение уве-

[1] *Бит фрагментации пакетов.* — Прим. ред.

личено на единицу. Каждый маршрутизатор на маршруте в описанном режиме отвечает сообщением об ошибке, что позволяет локальному маршрутизатору определить последовательные транзитные переходы.

В результате traceroute-запроса отображаются описанные ниже поля.

- Последовательный номер запроса — текущее число переходов.
- Имя узла текущего маршрутизатора.
- IP-адрес текущего маршрутизатора.
- Скоростные задержки (в миллисекундах) каждого из запросов в наборе.
- * — истекло время ожидания запроса.
- U — получено сообщение "порт недоступен".
- H — получено сообщение "узел недоступен".
- P — получено сообщение "протокол недоступен".
- N — получено сообщение "сеть недоступна".
- ? — получен пакет неизвестного типа.
- Q — получено сообщение о подавлении трафика отправителя.

Запросы трассировки маршрута будут отправляться до тех пор, пока не будет превышено максимальное значение параметра TTL (стандартно — 30 для протокола IP) или пока выполнение команды на маршрутизаторе не будет прервано пользователем с помощью управляющей последовательности <Ctrl>+<Shift>+<6>.

Эту команду можно запустить без параметров. В таком случае коммутатор запросит у пользователя параметры, перечень которых приведен ниже.

- Протокол (стандартно — ip). Также в IOS-коммутаторе Catalyst 6000 и других коммутаторах третьего уровня могут использоваться ключевые слова **appletalk**, **clns** или **vines**.
- Тестируемый адрес.
- Адрес отправителя — IP-адрес интерфейса маршрутизатора. Если значение не задано, используется ближайший к получателю интерфейс.
- Числовое отображение (стандартно не используется — значение "no") — по умолчанию отображается как имя, так и IP-адрес каждого транзитного узла. Если используется значение "yes", то отображается только IP-адрес. Эта функция может оказаться полезной в ситуации, когда DNS-сервер недоступен.
- Время ожидания в секундах (стандартно — 3) — период времени ожидания ответа на пробу.
- Количество проб (стандартно — 3) — количество проб, отправляемых на каждом TTL-уровне (или в каждом транзитном переходе).
- Минимальное значение времени жизни (TTL; стандартно 1) — стандартное значение 1 может быть изменено для того, чтобы начать без учета уже известных переходов на маршрутизаторах.
- Максимальное значение времени жизни (TTL; стандартно 30) — максимальное количество исследуемых переходов. Трассировка маршрута прекращается по достижении данного числа транзитных переходов или получателя.

- Номер порта (стандартно 33434) — UDP-порт получателя запросов.
- Параметры `loose`, `strict`, `record`, `timestamp` и `verbose` стандартно не используются. `loose` — свободный маршрут от отправителя с адресами переходов, `strict` — строгий маршрут от отправителя с адресами переходов, `record` — запись маршрута с определенным количеством переходов, `timestamp` — запись временных меток на каждом транзитном маршрутизаторе и `verbose` — включает режим вывода подробных сведений. Параметр `record` может быть полезен при просмотре записей адресов маршрутизаторов, которые проходит пакет на сквозном маршруте.

Внимание!

Некоторые маршрутизаторы не отвечают корректно на traceroute-запросы. В таких случаях несколько или все отправленные запросы отмечаются на дисплее звездочками (*).

3. Использование трассировки второго уровня для обнаружения коммутаторов на маршруте (*необязательно*).

Система COS	`l2trace` *src-mac dest-mac [vlan] [detail]* или `l2trace` *src-ip dest-ip [detail]*
Система IOS	Нет

Внимание!

Команда `l2trace` в настоящее время доступна только в COS-коммутаторах семейств Catalyst 4000, 5000 и 6000. Ее можно использовать для трассировки маршрута, содержащего коммутаторы других типов. Однако такие коммутаторы неспособны интерпретировать и отвечать на COP-сообщения `l2trace`. Для транзитных переходов с такими коммутаторами отображается превышение времени ожидания `l2trace`.

Трассировка второго уровня осуществляется с MAC-адреса отправителя (*src-mac* — шестнадцатеричные пары, разделенные пробелами) на MAC-адрес получателя, *dest-mac*. В адресной таблице коммутатора должны присутствовать как отправитель, так и получатель. Кроме того, отправитель и получатель должны находиться в одной VLAN-сети. Если узлы принадлежат различным VLAN-сетям, можно указать номер интересующей сети (*vlan*). Ключевое слово `detail` позволяет отобразить дополнительную информацию о передающей среде порта коммутатора на каждом транзитном переходе маршрута.

Если MAC-адреса определить трудно, можно задать отправителя и получателя с помощью IP-адресов (*src-ip* и *dest-ip*). Однако оба узла должны присутствовать в ARP-таблице, чтобы их MAC-адреса могли быть найдены.

Пример использования функции трассировки пакетов

В коммутаторе Catalyst 4000 трассировка второго уровня осуществляется с отправителя `00-b0-d0-a0-c2-df` к получателю `00-10-a4-c6-b4-b7`. Эти узлы расположены в одной VLAN-сети и присутствуют в адресной таблице коммутатора.

Адрес отправителя находится на порту 2/12 локального коммутатора. Первый транзитный узел второго уровня имеет IP-адрес 192.168.1.16, где адрес получателя находится в адресной таблице для порта 3/1 данного коммутатора.

Следует заметить, что второй транзитный узел второго уровня является коммутатором с адресом 192.168.1.253, который был идентифицирован с помощью протокола CDP. Однако либо модель коммутатора, либо его операционная система не поддерживают протокол l2trace. В результате время ожидания трассировки второго уровня истекает, и ответ от следующего коммутатора с адресом 192.168.1.253 не возвращается.

Система COS
```
cat4000> (enable) l2trace 00-b0-d0-40-01-d1 00-10-a4-c5-b4-b7
Starting L2 Trace
2/12 : 192.168.1.16 : 3/1
l2trace:ge response from neigh 192.168.1.253
l2trace:no response from neigh 192.168.1.253
Error in l2trace.
Cat4000> (enable)
```

Система IOS Нет

Рассмотрим ситуацию, когда трассировка второго уровня осуществляется с коммутатора Catalyst 6000 с операционной системой COS. Все коммутаторы на маршруте к получателю поддерживают l2trace. Адрес отправителя — 00-05-9b-fb-b8-80, адрес получателя — 00-04-9b-57-3e-c0.

Во время первой трассировки отправитель найден на порту 6/3 локального коммутатора, 192.168.1.4. Первый транзитный переход второго уровня начинается с портов 1/1-2, 2/1-2 локального коммутатора (четырехпортовый банк каналов FiberChannel). Второй транзитный переход — вход на портах 3/1-2, 4/1-2 (другой конец FEC-банка) коммутатора 192.168.1.252. Третий транзитный переход начинается на порту 3/3 коммутатора 192.168.1.252 и заканчивается на порту 7/1 коммутатора 192.168.1.7. Четвертый транзитный переход — переход к получателю на порту 7/3 коммутатора 192.168.1.7.

Далее осуществляется такая же трассировка второго уровня с добавлением ключевого слова **detail**. Следует отметить, как в данном случае показана дополнительная информация на каждом переходе: аппаратная платформа, имя узла (если оно доступно) и IP-адрес каждого транзитного коммутатора. Кроме того, показан каждый канал к следующему транзитному узлу с указанием номера порта и типа передающей среды.

Система COS
```
Cat6000-A (enable) l2trace 00-05-9b-fb-b8-80 00-04-9b-
57-3e-c0
Starting L2 Trace
6/3 : 192.168.1.4 : 1/1-2 2/1-2
3/1-2,4/1-2 : 192.168.1.252 : 3/3
7/1 : 192.168.1.7 : 7/3
Cat6000-A (enable)

Cat6000-A (enable) l2trace 00-05-9b-fb-b8-80 00-04-9b-
57-3e-c0 detail
Starting L2 Trace

l2trace vlan number is 901

00-05-9b-fb-b8-80 found in WS-C6509 named Cat6000-A on
port 6/1 1000MB full duplex
```

```
NS-C6509 . Cat6000-A . 192.168.1.4. 6/3 1000MB full
duplex -> 1/1 2,2/1-2 1000MB full duplex
NS-C6509 : : 192.168.1.252. 3/1-2,4/1-2 1000MB full
duplex -> 3/1 1000MB full duplex
WS-C6506 : : 192.168.1.7: 7/1 1000MB full duplex ->7/3
1000MB full duplex
Destination 00-04 9b 57-9c-c0 found on NS-C6509 on port
7/3 1000MB full duplex
Cat6000-A
(enable)
```

Система IOS	Нет

Дополнительная литература

Рекомендуемые ниже источники предоставляют дополнительную информацию по темам, рассматриваемым в этой главе.

Коттеди Кларк, Кевин Гамильтон, *Принципы коммутации в локальных сетях Cisco*. ИД "Вильямс", 2003.

Karen Webb, *Building Cisco Multilayer Switched Networks*, Cisco Press.

Tim Boyles and David Hucaby, *CCNP Switching Exam Certification Guide*, Cisco Press.

David Hucaby Steve McQuerry, *Cisco Field Manual: Router Configuration*, Cisco Press.

Maggora, Elliott, Pavone, Phelps and Thompson, *Performance and Fault Management*, Cisco Press.

В этой главе...

- **13.1: теоретические основы механизмов обеспечения качества обслуживания.** В разделе обсуждаются различные операции и механизмы, которые в целом составляют *качество обслуживания (Quality Of Service — QoS)*.
- **13.2: Конфигурирование средств QoS.** В разделе описываются этапы конфигурирования и мониторинга QoS-параметров в коммутаторах Catalyst.
- **13.3: экспорт данных QoS.** В этом разделе приведено описание конфигурационных этапов сбора и отправки статистической QoS-информации на внешние накопительные устройства.
- **13.4: управление QoS-функциями.** В разделе описывается методика конфигурирования коммутаторов, направленная на обеспечение поддержки таких административных протоколов, как *общая открытая служба правил (Common Open Policy Service — COPS)* и *протокол резервирования ресурсов (Resource Reservation Protocol — RSVP)*.

Глава 13

Качество обслуживания

13.1: теоретические основы механизмов обеспечения качества обслуживания

- Качество обслуживания (Quality of Service — QoS) определяет алгоритмы (policies), согласно которым коммутаторы и маршрутизаторы осуществляют доставку различных типов трафика. *QoS-домен (QoS domain)* — полная совокупность сетевых устройств, администрирование которых осуществляется с соблюдением требований QoS-стратегий.
- Для того чтобы гарантировать, что QoS-требования удовлетворяются, необходимо настроить сквозное обеспечение качества обслуживания на всех коммутаторах и маршрутизаторах в сети.
- Трафик необходимо классифицировать на границах QoS-домена. Там, где это возможно, трафик следует классифицировать как можно ближе к отправителю. Классификация может осуществляться на втором или третьем уровне в зависимости от системы функций, доступных на данной границе сети.
- На рис. 13.1 иллюстрируются QoS-операции, выполняемые на коммутаторе Catalyst. Ниже приводится их описание.
  - **Классификация (classification)** — выбор определенного трафика, к которому может применяться QoS-стратегия. Значениям приоритета входящих фреймов также можно доверять или классифицировать их повторно.
  - **Регулировка (policing)** — ограничение полосы пропускания, используемой каким-либо потоком данных. Алгоритмы, обеспечивающие регулировку (или регулировщики — policers), способны контролировать агрегированные или индивидуальные потоки данных, а также могут маркировать или отбрасывать трафик.
  - **Маркирование (marking)** — присвоение каждому фрейму значения DSCP (*Differentiated Services Code Point — точка кодировки дифференцированных служб*) на третьем уровне, *класса обслуживания (Class Of Service - CoS)* на втором уровне либо обоих значений.
  - **Планирование (scheduling)** — установка потока данных в определенную очередь порта коммутатора для исходящего либо входящего трафика.

- **Предотвращение перегрузок** — резервирование полосы пропускания в очередях портов коммутатора. Трафик, превышающий пороговое значение, может быть отброшен или значение его приоритета может быть понижено, что позволит освободить пространство для другого трафика в очередях.

- Все QoS-операции коммутаторов Catalyst основаны на понятии *внутреннего DSCP-значения (internal DSCP)*. Это значение передается системным доверия (trust state) порта входа и проходит весь QoS-процесс с каждым фреймом. После выхода внутреннее DSCP-значение может использоваться для маркирования других QoS-значений внутри данного фрейма. На рис. 13.1 иллюстрируются внутренние DSCP-операции.

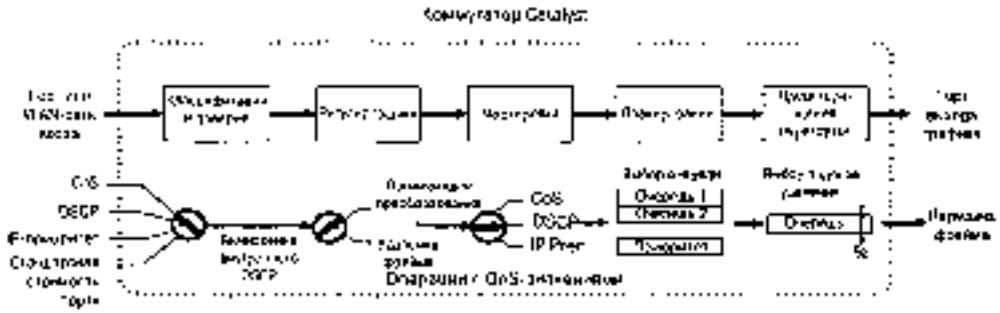


Рис. 13.1. QoS-операции коммутаторов Catalyst с внутренним DSCP-значением

В табл. 13.1 перечислены возможности обеспечения QoS в различных коммутаторах семейства Cisco Catalyst.

QoS-классификация второго уровня и маркирование

На втором уровне не существует механизма для указания приоритета или важности содержимого обычных фреймов. Следовательно, доставка фреймов второго уровня должна осуществляться на основании методики негарантированной доставки (best effort).

В то же время, если *виртуальные локальные сети* (VLAN) объединены с помощью одного магистрального канала, то этот канал предоставляет возможность транспортировать сведения о приоритете вместе с каждым фреймом. Ниже представлено описание средств транспортировки CoS параметров второго уровня.

- **Магистральный канал 802.1Q IEEE.** Фреймы маркируются двенадцатибитовым идентификатором VLAN-сети (VLAN ID). CoS значение содержится в трех битах приоритета протокола 802.1 в поле User (пользовательские данные). Фреймы в собственной VLAN-сети не маркируются вообще, для них задано стандартное CoS-значение или приоритет для порта коммутатора. На рис. 13.2 демонстрируется формат метки 802.1Q-инкапсуляции.

0	1	2	3
Метка протокола (0x8100)		Пользовательские данные	Идентификатор VLAN-сети (12 битов)

Рис. 13.2. Формат 802.1Q-инкапсуляции магистрального канала

Таблица 13.1. QoS-возможности коммутаторов Cisco Catalyst

Семейство коммутаторов	Классификация	Регулировка	Планирование	Предотвращение перегрузок	Маркирование
6000 с платой PFC (Policy Feature Card) или PFC2	На основании CoS, DSCP, IP-приоритета, списков доступа (уровни 3 и 4)	Агрегированные и микропотоки (1023 регулировщика)	Конфигурируемые очереди	Конфигурируемое с порогами удаления	CoS, DSCP
6000 без платы PFC	На основании CoS, MAC-адресов	Нет	Конфигурируемые очереди	Конфигурируемое с порогами удаления	CoS
4000 с модулем Supervisor III	На основании CoS, DSCP, IP-приоритета, списков доступа (уровни 3 и 4)	Агрегированные и микропотоки (1023 регулировщика)	Конфигурируемые очереди	Конфигурируемое с порогами удаления	CoS, DSCP
4000 без Supervisor III	На основании CoS, MAC-адресов	Нет	Конфигурируемые очереди	Фиксированные пороги	Стандартный класс CoS
5000	На основании CoS, MAC-адресов	Нет	Фиксированное	Конфигурируемое с порогами удаления	CoS
3550	На основании CoS, DSCP, IP-приоритета, списков доступа (уровни 3 и 4)	Агрегированные и микропотоки (128 регулировщика)	Конфигурируемые очереди	Конфигурируемое с порогами удаления	CoS, DSCP
3500XL/2900XL	На основании CoS	Нет	Фиксированное	Нет	CoS

- **Магистральный канал ISL** *(Inter-Switch Link — межкоммутаторный канал).* Фреймы маркируются пятнадцатибитовым идентификатором VLAN-сети. CoS-значение содержится в трех младших битах поля User (пользовательские данные). Коммутаторы Catalyst копируют CoS-биты стандарта 802.1 из фрейма в магистральном канале 802.1Q в поле пользовательских данных фреймов ISL-канала, хотя эта функция не стандартизирована. На рис. 13.3 проиллюстрирован формат ISL-метки.

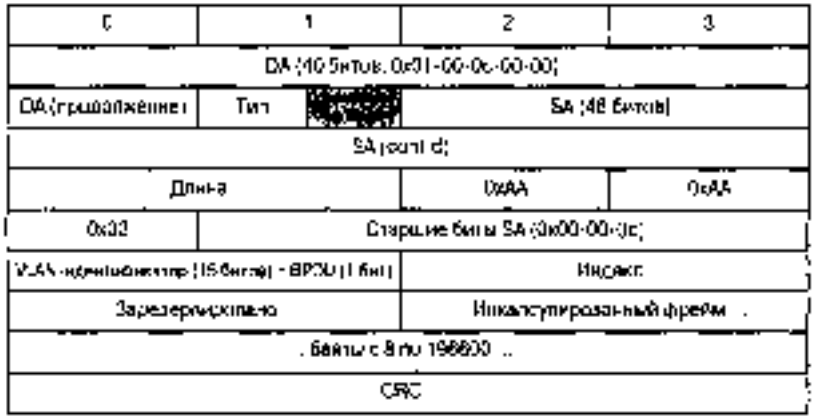


Рис. 13.3. Формат ISL-инкапсуляции магистрального канала

QoS-классификация третьего уровня и маркирование

Качество обслуживания также основывается на понятии *дифференцированной службы (Differentiated Service – DiffServ)*, в которой QoS-спецификация транспортируется внутри каждого пакета третьего уровня. В IP-пакетах имеется байт *типа обслуживания (Type of Service — ToS)*, который форматируется согласно верхней строке рис. 13.4. Биты P2, P1 и P0 формируют значение IP-приоритета. ToS-значение формируется из битов T3, T2, T1 и T0.

Для DiffServ тот же байт называется байтом *дифференцированных служб (Differentiated Services — DS)* и форматируется согласно нижней строке рис. 13.4. Биты DS5-DS0 формируют *точку кодирования дифференцированных служб (Differentiated Services Code Point — DSCP)*. DSCP-точка устанавливается в целях обратной совместимости с битами IP-приоритета, поскольку две указанные величины совместно используют один и тот же байт в IP-заголовке.

ToS-байт:	P2	P1	P0	T3	T2	T1	T0	Ноль
DS-байт	DS5	DS4	DS3	DS2	DS1	DS0	ECN1	ECN0
	(Селектор класса)			(Приоритет удаления)				

Рис. 13.4. Форматы ToS- и DSCP-байтов

Биты DS5, DS4 и DS3 формируют селектор класса DSCP. Классы с 1 по 4 называются уровнями служб *гарантированной пересылки (Assured Forwarding — AF)*.

Более высокие номера класса указывают на высокоприоритетный трафик. В каждом классе или уровне AF-служб имеются три уровня *приоритета уничтожения пакета (drop precedence)*.

- низкий (1);
- средний (2);
- высокий (3).

Трафик в AF-классах может удаляться с наибольшей вероятностью в самой низкой категории и с наименьшей — в высокой категории. Иными словами, трафик уровня 4 класса AF-службы с приоритетом удаления 3 будет доставлен раньше, чем трафик класса 4 AF-службы с приоритетом удаления 1, который, в свою очередь, доставляется раньше трафика AF-класса 3 с приоритетом удаления 3, и так далее.

Класс 5 также называется классом *ускоренной доставки (Expedited Forwarding — EF)* и обеспечивает наивысшее качество обслуживания и минимальную вероятность удаления пакетов. Селектор *стандартного (Default)* класса (DSCP 000 000) обеспечивает только негарантированную пересылку данных.

Оба класса 6, *межсетевое управление (Internetwork Control)*, и 7, *сетевое управление (Network Control)*, зарезервированы для сообщений управления сетью, включающих в себя трафик протокола распределенного связующего дерева и протоколов маршрутизации, т.е. трафик, который в сети генерируется пользователями и обычно рассматривается как высокоприоритетный.

В табл. 12.2 описано преобразование имен и битов IP-приоритета в DSCP-значения. DSCP-значения разделяются по *индивидуальному режиму (Per-Hop Behavior — PHB)*, селектору класса и приоритету удаления. Нередко на DSCP-значения ссылаются по именам точки кодирования (например, AF32), которые также перечислены в данной таблице. DSCP-биты показаны наряду с их десятичными эквивалентами. Во многих командах, связанных с DSCP, необходимо вводить десятичные DSCP-значения, даже если трудно связать десятичные числа с соответствующими уровнями DSCP-служб и функции PHB. Приведенная ниже таблица может служить в качестве удобного справочника.

Совет

Классы CoS второго уровня и коды DSCP/ToS третьего уровня являются полностью независимыми понятиями. Как таковые, два QoS-значения не смешиваются и не переводятся друг в друга автоматически. Преобразование значений CoS-DSCP на границе второго и третьего уровней должно быть обеспечено коммутатором.

DSCP/ToS-значения третьего уровня транспортируются внутри каждого IP-пакета, что позволяет QoS-информации распространяться автоматически. CoS-значение второго уровня не содержится во фреймах второго уровня и может транспортироваться только через магистральный канал. Для распространения CoS-значений необходимо использовать магистральный канал между коммутаторами.

Таблица 13.2. Преобразование полей IP-приоритета и DSCP

IP-приоритет (3 бита)			DSCP (6 битов)				
Имя	Значение	Биты	PHB-режим	Селектор класса	Приоритет удаления	Имя точки кодирования	DSCP-биты (десятичные значения)
Регулярный (Routine)	0	000	Стандартный			Стандартное	000 000 (0)
Приоритетный (Priority)	1	001	AF	1	1:Низкий	AF11	001 010 (10)
					2:Средний	AF12	001 100 (12)
					3:Высокий	AF13	001 110 (14)
Мгновенный (Immediate)	2	010	AF	2	1:Низкий	AF21	010 010 (18)
					2 Средний	AF22	010 100 (20)
					3:Высокий	AF23	010 110 (22)
Кратковременный (Flash)	3	011	AF	3	1:Низкий	AF31	011 010 (26)
					2 Средний	AF32	011 100 (28)
					3 Высокий	AF33	011 110 (30)
Кратковременный переустанавливаемый (Flash Override)	4	100	AF	4	1:Низкий	AF41	100 010 (34)
					2 Средний	AF42	100 100 (36)
					3 Высокий	AF43	100 110 (38)
Критический (Critical)	5	101	EF	Нет	Нет	EF	101 110 (46)[1]
Межсетевое управление (Internetwork Control)	6	110	Нет	Нет	Нет	Нет	Нет[2]
Сетевое управление (Network Control)	7	111	Нет	Нет	Нет	Нет	Нет[2]

[1] *Значение IP-приоритета 5 (DSCP EF) соответствует диапазону DSCP-битов с 101000 по 101111, или 40-47. Однако широко используется только значение 101110, или 46, которое получило обозначение EF.*

[2] *Значения IP-приоритетов 6 и 7 занимают DSCP-диапазоны 48-55 и 56-63 соответственно. Однако эти значения обычно используются протоколами управления сетью и для простоты не включены в таблицу.*

Очереди в коммутаторах Catalyst

В портах коммутаторов Catalyst имеются как входящие (ingress), так и исходящие (egress) очереди, в которых буферизируются фреймы по мере их получения или перед отправкой. В каждом порту обычно имеются несколько очередей, каждая из которых сконфигурирована для относительного приоритета трафика. Например, очередь с наименьшим приоритетом обслуживается только после высокоприоритетных очередей.

В большинстве коммутирующих платформ имеется очередь строгого (strict-priority) приоритета, которая используется для трафика, чувствительного ко времени доставки (time-critical traffic). Эта очередь всегда обслуживается перед любой другой в порту.

В каждой очереди обычно имеется одно или несколько пороговых значений, которые указывают, можно ли отбрасывать трафик. Когда уровень заполнения очереди меньше порогового значения, фреймы не отбрасываются. Если заполнение очереди превышает порог, то вероятность отбрасывания возрастает.

В процессе QoS-конфигурирования необходимо ссылаться на очереди по номерам. Стандартной очередью с наименьшим приоритетом всегда является очередь 1. Номера последующих очередей с более высоким приоритетом начинаются с 2. Очередь строгого приоритета всегда получает наивысший индексный номер.

Порты коммутаторов Cisco Catalyst описываются с помощью обозначения очередей: *xpyq*t, где

- **p** — число очередей строгого приоритета, задается значением *x*;
- **q** — число стандартных очередей, задается значением *y*;
- **t** — число конфигурируемых порогов для каждой очереди, задается значением *z*.

Например, порт типа **1p3q1t** имеет одну очередь строгого приоритета, три стандартные очереди и четыре пороговых значения для каждой очереди. Стандартная низкоприоритетная очередь имеет номер 1, тогда как очередь строгого приоритета имеет номер 2.

13.2: конфигурирование средств QoS

- QoS операции и правила могут применяться на основе описанных ниже объектов:
  - **На основе портов (port-based).** Все данные проходят через определенный порт. Такая методика обычно используется в коммутаторах с блоком коммутации третьего уровня.
  - **На основе VLAN-сетей (VLAN-based).** Все данные проходят через определенную VLAN-сеть на коммутаторе. Обычно эта методика используется в коммутаторах с блоком коммутации второго уровня или когда QoS-стратегии являются общими для всего трафика в какой-либо VLAN-сети.
- Классификация может осуществляться на входящих портах коммутатора. Входящим значениям CoS, IP-приоритета и DSCP можно доверять, принимая значения, заданные подключенным устройством. Такой подход является приемлемым, если отправитель значений известен и находится под административным контролем. Если значениям при входе в коммутатор нельзя доверять, то они могут быть преобразованы в новые значения. Внутреннее DSCP-значение происходит из классификации для каждого фрейма.

- Для поддержки расширенных потребностей QoS можно точно настроить очереди и планировку для входного порта коммутатора.
- Для управления входящим трафиком могут применяться регулировщики потоков.
  - В регулировщиках потоков (policers) для мониторинга использования полосы пропускания потоком трафика применяется алгоритм *маркерной ячейки (token bucket)*. Значения длины входящих фреймов по мере поступления добавляются к маркерной ячейке. Каждые 0,25 миллисекунд (1/4000 секунды) значение *согласованной скорости передачи информации (Committed Information Rate — CIR)* или средняя согласованная скорость (average policed rate) вычитается из маркерной ячейки. Идея заключается в том, чтобы удержать маркерную ячейку равной нулю при стабильной скорости передачи данных (sustained data rate).
  - Регулировщик потоков допускает определенные превышения скорости трафика по сравнению со средней скоростью. Некоторое превышение скорости при всплесках допускается, когда маркерная ячейка достигает уровня всплеска (в байтах). Такой трафик называется *профильным (in-profile traffic)*.
  - Когда размер маркерной ячейки превышает значение всплеска, регулировщик рассматривает этот поток трафика как чрезмерный. С помощью модуля PFC2 можно определить *пиковую скорость передачи информации (Peak Information Rate — PIR)*. Когда потоки данных превышают максимальный размер всплеска по сравнению с PIR, регулировщик рассматривает поток как находящийся в состоянии нарушения. Такой тип трафика называется *непрофильным (out-of-profile traffic)*.
  - Агрегированные регулировщики отслеживают и контролируют кумулятивный поток, который проходит через один или несколько портов ввода или VLAN-сетей. В коммутаторе Catalyst 6000 можно определить до 1023 агрегированных регулировщиков.
  - Регулировщики микропотока осуществляют мониторинг и контроль одного определенного потока данных, или *микропотока (microflow)*. IP-микропоток определяется IP-адресами отправителя и получателя, протоколом четвертого уровня, а также номерами портов отправителя и получателя. В IPX-микропотоке имеются общие сети отправителя и получателя, а также общий узел-получатель. В микропотоке MAC-уровня имеется общий приемник, а также общие MAC-адреса отправителя и получателя. В коммутаторе Catalyst 6000 можно определить до 63 регулировщиков микропотоков.
- С помощью *записей списков доступа (Access Control Entries — ACE)* применяется соответствие трафика на основании адреса и информации порта четвертого уровня. ACE-записи группируются в *списки управления доступом (Access Control Lists — ACL)*, или QoS-стратегии, которые применяются к определенным портам коммутатора.
- Предотвращение перегрузок конфигурируется путем присвоения пороговых значений различным исходящим очередям. Трафик отбрасывается, если уровень очереди превышает соответствующий порог, что позволяет резервировать пространство в очереди для другого трафика.
- Чтобы назначить классы трафика очередям и портам с относительными приоритетами обслуживания, нужно точно настроить распределение исходящих портов коммутатора.

Совет

QoS-операции в коммутаторах Catalyst 2900XL и 3500XL ограничены доверительностью QoS-параметров и алгоритмом планирования фиксированной очереди. Поэтому чижи данные коммутаторы описываются отдельно

1. Классификация трафика на основе порта (*необязательно*).

   а) Установка стандартного входного CoS-значения (*необязательно*).

Система COS	Нет
Система IOS	`switchport priority default cos` (режим конфигурирования интерфейса)

   Немаркированные фреймы получают CoS-значение, заданное параметром *cos* (0-7).

   б) Не доверять любой входящей информации (*необязательно*).

Система COS	Нет
Система IOS	`switchport priority override` (режим конфигурирования интерфейса)

   В коммутаторах Catalyst 3500XL CoS-значение устанавливается равным стандартному CoS-значению, сконфигурированному на этапе 1а. Стандартно все порты коммутатора заменяют входящие немаркированные или CoS-значения статического доступа нулем.

   в) Передача инструкции подсоединенному прибору по обработке CoS-значения (*необязательно*).

Система COS	Нет
Система IOS	`switchport priority extend {cos cos | none | trust}` (режим конфигурирования интерфейса)

   CoS-доверие может распространяться на IP-телефон Cisco или другой прибор, подключенный к порту коммутатора Catalyst 3500XL. Коммутатор может передать данному прибору инструкцию о том, как доверять CoS-значениям от других подключенных к нему устройств. CoS-доверие может быть задано с помощью ключевого слова **cos** (изменяет CoS во фреймах от других устройств на значение *cos*, от 0 до 7), **none** (стандартная установка, прибор не предпринимает никаких действий по отношению к CoS-значению) или **trust** (прибор доверяет и пересылает CoS-значения во фреймах от других устройств). Более подробно конфигурирование IP-телефона рассматривается в главе 4, "Поддержка передачи голосовых данных".

2. Планирование очереди порта

   - Коммутаторы Catalyst 2900XL и 3500XL имеют одну входящую очередь. Ее конфигурирование невозможно.

- Эти коммутаторы имеют порты выхода типа 2q0t. Фреймы с CoS-значениями от 0 до 3 назначается низкоприоритетной очереди (очередь 1). Фреймы с CoS-значениями от 4 до 7 направляются высокоприоритетной очереди (очередь 2).
- Планирование исходящих очередей не является конфигурируемым. Кроме того, пороги предотвращения перегрузок зафиксированы на уровне 100 процентов.

Остальные конфигурационные параметры коммутатора Catalyst

Внимание!

Некоторые этапы конфигурирования применимы только к коммутаторам с блоками коммутации второго или третьего уровня. Такие этапы отмечаются следующим образом:

только для коммутаторов второго уровня — коммутатор Catalyst 6000 без платы PFC или PFC2, коммутатор Catalyst 5000;

только для коммутаторов третьего уровня — коммутатор Catalyst 6000 с платой PFC или PFC2, коммутатор Catalyst 4000 с блоком Supervisor III и коммутатор Catalyst 3550.

Приведенные команды операционной системы IOS часто начинаются ключевым словом `mls`. Хотя эти команды поддерживаются системой IOS блока Supervisor III коммутатора Catalyst 4000, их синтаксис идентичен, но ключевое слово `mls` опущено.

1. Включение QoS-функций.

Система COS	`set qos {enable | disable}`
Система IOS	`mls qos` (режим глобальной конфигурации)

Стандартно функции QoS отключены. Весь трафик коммутируется в сквозном режиме, при котором обеспечивается только гарантированная доставка.

2. Применение QoS-функций к портам или VLAN-сетям (*только для коммутаторов третьего уровня*).

Система COS	`set port qos` *mod/port* `{port-based | vlan-based}`
Система IOS	`mls qos vlan` (режим глобальной конфигурации)

Стандартно качество обслуживания основано на портах (режим `port-based`; команда по `mls qos vlan-based`) или применяется к отдельным портам второго уровня. QoS-правила вместо этого могут применяться ко VLAN-сети порта. При применении приложения любые QoS-правила, основанные на портах, с этого порта снимаются.

3. Классификация трафика в зависимости от порта.

Совет

На этом этапе порт коммутатора можно настроить так, чтобы выбранные входные QoS-параметры были доверяемыми. В противном случае QoS-стратегия может быть определена так, чтобы QoS-параметры были доверяемыми при соблюдении некоторых условий.

Такие настройки описаны в этапах 6 (COS) и 8 (IOS). В IOS-коммутаторе состояние доверия может быть задано только на физических портах, а не на VLAN-интерфейсах.

а) Установка входящего CoS-значения (*необязательно*).

Система COS	`set port qos` *mod/ports* `cos` *cos-value*
Система IOS	`mls qos cos` *cos-value* (режим конфигурирования интерфейса)

Для фреймов, полученных на портах, параметры которых не доверяют, а также для немаркированных фреймов, полученных на портах, которым доверяют (фреймы в собственной 802.1Q VLAN-сети), CoS-значение устанавливается равным параметру *cos-value* (от 0 до 7, стандартно 0).

б) Не доверять любой входящей информации (*необязательно*).

Система COS	`set port qos` *mod/port* `trust untrusted`
Система IOS	`no mls qos trust` (режим конфигурирования интерфейса)

Входящие значения CoS, DSCP и IP-приоритета являются теми, которым не доверяют. Все они повторно классифицируются на основе каких либо соответствующих QoS-правил или таблиц преобразования. Если стратегии отсутствуют, то оба значения — как CoS, так и DSCP — устанавливаются равными нулю.

При включенных QoS-функциях стандартным для каждого порта является состояние `untrusted`.

в) Расширение доверительной границы к IP-телефону (*необязательно, только для коммутаторов Catalyst 6000*)

- Установка доверия на порту доступа телефона

Система COS	`set port qos` *mod/ports* `trust-ext {trusted untrusted}`
Система IOS	Нет

IP-телефон Cisco имеет собственный порт коммутатора уровня доступа к среде, к которому может подключаться PC-станция. Стандартно состояние такого порта — `untrusted`, что приводит к установке нулевых значений CoS и IP-приоритета для входящих фреймов. Для того чтобы разрешить персональному компьютеру маркировать собственные пакеты значениями IP-приоритета, следует установить режим `trusted`.

- Установка стандартного CoS-значения для порта доступа телефона.

Система COS	`set port qos` *mod/ports* `cos-ext` *cos-value*
Система IOS	Нет

Когда порт доступа телефона переводится в режим `untrusted`, телефон для всех входящих фреймов установит CoS-значение равным параметру *cos-value* (от 0 до 7; стандартно 0). Более подробная информация об IP-телефонах приведена в главе 14, "Поддержка передачи голосовых данных".

т) Стандартно доверять входящему CoS-значению (*необязательно, только для коммутаторов третьего уровня*).

Преобразование CoS-значений во внутренние DSCP-значения.

Система COS	`set qos cos-dscp-map` *dscp1 ... dscp8*
Система IOS	`mls qos map cos-dscp` *dscp1 ... dscp8* (режим глобальной конфигурации)

CoS-значения (от 0 до 7) из входящих фреймов преобразовываются в соответствующие восемь DSCP-значений (*dscp1* по *dscp8*, определенные значения с шагом 8 в диапазоне от 0 до 63). Таким результирующие внутренние DSCP-значения используются QoS-процессами в коммутаторе. Стандартное преобразование показано ниже.

CoS	DSCP
0	0 (негарантированная доставка)
1	8 (AF-класс 1, негарантированная доставка)
2	16 (AF-класс 2, негарантированная доставка)
3	24 (AF-класс 3, негарантированная доставка)
4	32 (AF-класс 4, негарантированная доставка)
5	40 (EF-класс, негарантированная доставка)
6	48 (межсетевое управление, негарантированная доставка)
7	56 (сетевое управление, негарантированная доставка)

Стандартно значения приоритета отбрасывания не используются, в результате чего образуются DSCP-значения, которые несколько отличаются от приведенных в табл. 13.2, поскольку все биты приоритета отбрасывания равны 000. При необходимости преобразовать в коммутаторе CoS-значения в DSCP следует изменить стандартное преобразование так, чтобы вместо них использовались отличные приоритеты отбрасывания. Для возврата к стандартному преобразованию используется команда `clear qos cos-dscp-map` или по `mls qos map cos-dscp`.

- Включение CoS-доверия на одном или нескольких портах.

Система COS	`set port qos` *mod/port* `trust trust-cos`
Система IOS	`mls qos trust cos` (режим конфигурирования интерфейса)

Доверять только входящему CoS-значению, от которого будут происходить ToS- или DSCP-значения.

Совет

В портах со скоростью 10/100 Мбит/с COS-коммутатора в состоянии `trust-cos` пороги отбрасывания включаются только при получении. Для установок на данных портах состояния доверия требуется также определить ACL-список проверки соответствия `trust-cos`. Необходимые для этого настройки описаны в этапе 6.

л) Установка CoS-значения в зависимости от MAC-адреса получателя *(необязательно; только для коммутаторов второго уровня)*.

Система COS	`set qos mac-cos` *dest-mac vlan cos*
Система IOS	Нет

Для всех фреймов, адресованных MAC-адресу *dest-mac* во VLAN-сети номер *vlan*, можно установить CoS-значение, равное параметру *cos* (от 0 до 7). Это может оказаться полезным для повторной классификации трафика, направленного к определенному узлу-получателю.

м) Стандартно доверять входящему значению IP-приоритета *(необязательно; только для коммутаторов третьего уровня)*.

- Преобразование IP-приоритета во внутреннее DSCP-значение

Система COS	`set qos ipprec-dscp-map` *dscp1 ... dscp8*
Система IOS	`mls qos map ip-prec-dscp` *dscp1 ... dscp8* (режим глобальной конфигурации)

Значения IP-приоритета (от 0 до 7 или `routine`, `priority`, `immediate`, `flash`, `flash-override`, `critical`, `internet` и `network`) из входящих пакетов преобразовываются в соответствующие весовые DSCP-значения (с *dscp1* по *dscp8*, от 0 до 63, стандартно 0, 8, 16, 24, 32, 40, 48 и 56). Затем полученные внутренние DSCP-значения используются QoS-функциями. В приведенной ниже таблице иллюстрируется стандартное преобразование.

ToS	DSCP
0 (*routine* — *регулярный*)	0 (*негарантированная доставка*)
1 (*priority* — *приоритетный*)	8 (AF-класс 1, негарантированная доставка)
2 (*immediate* — *немедленный*)	16 (AF-класс 2, негарантированная доставка)
3 (*flash* — *быстрый*)	24 (AF-класс 3, негарантированная доставка)
4 (*flash-override* — *быстрый переустанавливаемый*)	32 (AF-класс 4, негарантированная доставка)
5 (*critical* — *критический*)	40 (EF-класс, негарантированная доставка)
6 (*internet* — *межсетевой*)	48 (межсетевое управление, негарантированная доставка)
7 (*network* — *сетевой*)	56 (сетевое управление, негарантированная доставка)

Значения приоритета отбрасывания стандартно не используются, в результате чего возникают DSCP-значения, которые несколько отличаются от приведенных в табл. 13.2, поскольку все биты приоритета отбрасывания равны 000. При необходимости преобразовать в коммутаторе CoS-значения в код DSCP следует изменить стандартное преобразование так, чтобы вместо них использовались отличные приоритеты отбрасывания. Для возврата к стандартному преобразованию используется команда `clear qos ipprec-dscp-map` или `no mls qos map ip-prec-dscp`

- Доверие IP-приоритету на одном или нескольких портах.

Система COS	`set port qos mod/port trust trust-ipprec`
Система IOS	`mls qos trust ip-precedence` (режим конфигурирования интерфейса)

- Доверять только входящему значению IP-приоритета (ToS), от которого будут происходить DSCP-значения.

ж) Стандартно доверять входящему DSCP-значению (*необязательно, только для коммутаторов третьего уровня*)

Система COS	`set port qos mod/port trust trust-dscp`
Система IOS	`mls qos trust dscp` (режим конфигурирования интерфейса)

Можно задать такие настройки, что значением, которому доверяют, будет только входящее DSCP значение, при этом ToS и DSCP значения останутся неизменными. При другом преобразовании внутренние DSCP-значения не применяются.

з) Преобразование DSCP-значений между QoS доменами (*только для коммутатора Catalyst 3550*).

- Создание таблицы изменения DSCP-значений

Система COS	Нет
Система IOS	`mls qos map dscp-mutation` *dscp-mutation-name* *in-dscp* `to` *out-dscp* (режим глобальной конфигурации)

Если порт коммутатора расположен на границе QoS-домена, то входящие DSCP-значения могут быть преобразованы в группу других DSCP-значений. Таблица изменения (mutation map) с именем, заданным с помощью параметра *dscp-mutation-name* (текстовая строка), содержит входящие значения *in-dscp* (до 8 значений от 0 до 63, разделенных пробелами), которые преобразовываются в соответствующие новые значения *out-dscp* (до 8 значений от 0 до 63, разделенных пробелами). Если требуется преобразовать более 8 значений, эту команду можно повторить.

- Применение таблицы изменений к интерфейсу.

Система COS	Нет
Система IOS	`mls qos dscp-mutation` *dscp-mutation-name* (режим конфигурирования интерфейса)

Стандартно на интерфейсе не происходит каких-либо изменений. В противном случае используется таблица изменения с именем, заданным с помощью параметра *dscp-mutation-name* (текстовая строка). Для каждого Gigabit Ethernet-интерфейса может существовать отдельная таблица изменения, тогда как для каждой группы, состоящей из 12 Ethernet-интерфейсов 10/100, может использоваться только одна такая таблица.

4. Точная настройка очередей портов входа (*необязательно*).

Совет

Стандартно в портах входа используется предотвращение перегрузок и планирование, описанное в табл. 13.3

Таблица 13.3. Предотвращение перегрузок и планирование для входных портов

Тип очереди	Количество порогов (стандартная очередь) CoS: процент отбрасывания последних пакетов (tail-drop) низкий или высокий WRED							
	T1	T2	T3	T4	T5	T6	T7	T8
1q4t	0,1,50%	2,3,60%	4,5,80%	6,7,100%				
1p1q4t	0,1,50%	2,3,0%	4,80%	6,7,100%				
1p1q8t	0:	1:	2	3:	4:	6.	7:	
	40%/70%	40%/70%	50%/80%	50%/80%	60%/90%	60%/90%	70%/100%	

Совет

Порты всех типов устанавливают фреймы с CoS-значением 5 в свои очереди строгого приоритета (кроме порта 1q4t, который ее не имеет). Порты 1p1q0t не имеют пороговых значений: все фреймы с CoS-значениями, отличными от 5, назначаются стандартной очереди и отбрасываются, когда эта очередь заполнена на 100 процентов

а) Точная настройка коэффициента входящей очереди (*необязательно*).

Система COS	`set qos rxq-ratio port-type queue1 queue2`
Система IOS	`rcv-queue queue-limit` *queue1 queue2* (режим конфигурирования интерфейса)

Внимание!

При использовании этой команды все порты проходят цикл отключения-включения каналов. В реально действующей сети это приводит к простоям, связанным с отключением портов и последующим прохождением ими STP-состояний.

Стандартная очередь (очередь 1) получает 80 процентов доступного пространства, тогда как очередь строгого приоритета (очередь 2) получает 20 процентов пространства. Если QoS-функции отключены, то стандартная очередь получает 100 процентов пространства.

Следует оценить соотношение обычного и приоритетного трафиков, поступающих в порт коммутатора. Чтобы установить процентную часть (1–99) для каждой из двух очередей приема, необходимо использовать значения *queue1* и *queue2*, которые в сумме должны составлять 100 процентов.

б) Установка порогов предотвращения перегрузок (*необязательно*).

- Использование стандартных очередей приема с отбрасыванием последних пакетов (*необязательно*)

Система COS	`set qos drop-threshold` *port-type* `rx queue` *queue-id threshold-percent 1 ... threshold percent n*
Система IOS	`rcv-queue threshold` *queue-id threshold-percent-1 ... threshold-percent-n* или `wrr-queue threshold` *queue-id threshold-percent-1 ... threshold-percent-n* (обе команды вводятся в режиме конфигурирования интерфейса)

Для большинства очередей приема порта коммутатора (1q4t, 1p1q4t, 2q2t и 1p1q8t) можно применять стандартное предотвращение перегрузок с отбрасыванием последних пакетов. Стандартно фреймы с CoS-значением 5 назначаются очереди строгого приоритета. Все остальные фреймы назначаются стандартной очереди.

Количество доступных в очереди пороговых значений — число, предшествующее параметру t в обозначении типа очереди. Для каждого порога можно назначить процент буферного пространства, которое доступно для принимаемых фреймов. Значения *threshold percentage n* (от 1 до 100 процентов) задаются в последовательном порядке. Если размер буфера превышает уровень порогового значения, то новые входящие фреймы отбрасываются.

Совет

В IOS-коммутаторе очередь 1q4t обслуживается с помощью взвешенного циклического (Weighted Round-Robin — WRR) алгоритма. Следовательно, пороговые значения необходимо устанавливать, используя команду `wrr-queue threshold`.

- Использование очереди приема со *взвешенным произвольным ранним обнаружением (Weighted Random Early Detection — WRED)* (*необязательно*).

Система COS	`set qos wred` *port-type* `queue` *queue-id* *{thr1-min}/{thr1-max} {thr2-min}/{thr2-max} ...*
Система IOS	`rcv-queue random-detect min-threshold` *queue-id thr1-min thr2-min ...* `rcv-queue random-detect max-threshold` *queue-id thr1-max thr2-max ...* (обе команды вводятся в режиме конфигурирования интерфейса)

Для портов типа 1p1q8t используется метод WRED. Идентификатор очереди (*queue-id*) равен единице (стандартная очередь) или двум (приоритетная

очереди. Для каждого из восьми пороговых значений используются два ограничения: минимум (*thrl-min* от 1 до 100 процентов) и максимум (*thrl-max* от 1 до 100 процентов).

Если размер буфера ниже минимального уровня, фреймы не отбрасываются. По мере того как буфер превышает минимум, но не достигает максимума, вероятность отбрасывания фреймов возрастает. При превышении максимального уровня все фреймы отбрасываются.

Совет

Для очередей приема `7q8q` в IOS также требуется команда `wrr-queue random-detect` *queue-id*, активизирующая пороги отбрасывания WRED.

в) Настройка планирования и предотвращения перегрузок на входном интерфейсе *(необязательно)*.

Система COS	`set qos map` *port-type* `rx` *queue-id threshold-id cos cos-list*
Система IOS	`rcv-queue cos-map` *queue-id threshold-id cos-list* или `wrr-queue cos-map` *queue-id threshold-id cos-list* (обе команды вводятся в режиме конфигурирования интерфейса)

Если входящим CoS-значениям доверяют, исходя из настроек этапа 3c или QoS-стратегии, то фреймы с определенными CoS-значениями могут быть преобразованы и отправлены к определенным входящим очередям и портам. Параметр *cos-list* может принимать одно значение (от 0 до 7), множество значений, разделенных запятыми, или диапазон значений, заданный с помощью дефиса. Такое преобразование устанавливается для всех портов коммутатора (QoS-коммутатор) или для каждого интерфейса (IOS-коммутатор).

Параметр *port-type* — тип доступной очереди, который можно определить с помощью команды `show port capabilities` (COS) или `show queueing interface` (IOS). Параметры *threshold-id* (1-4) и *queue-id* (1 — для стандартной или 2 — для очереди строгого приоритета) идентифицируют определенное пороговое значение и очередь, в которую будут помещаться входящие фреймы. Диапазон значений зависит от аппаратного обеспечения порта коммутатора.

Стандартно, когда функции QoS включены, CoS-значения с 0 по 7 привязываются к стандартной входящей очереди (очередь 1). Если поддерживается очередь строгого приоритета (очередь 2, тип очереди начинается с `1p...`), то значение CoS 5 привязывается к ней.

5. Создание регулировщика для управления потоком входящих пакетов *(необязательно; только для коммутаторов уровня доступа)*.

а) Использование агрегированного регулировщика *(необязательно)*.

Система COS	`set qos policer aggregate` *aggregate_name* `rate` *rate* `burst` *burst* `{drop | policed-dscp} [erate` *peak-rate* `[policed-dscp]]`

Система IOS	`mls qos aggregate-policer` *aggregate-name rate burst* *[max-burst]* **[pir** *peak-rate***]** **[conform-action** *action***]** **[exceed-action** *action***]** **[violate-action** *action***]** (режим глобальной конфигурации)

В COS-коммутаторе регулировщик определяется средней или CIR-скоростью (`rate` 0 или от 32 до 8 000 000 Кбит/с), разрешено также превышение этой скорости (параметр `burst` от 1 до 32 000 Кб). Если используется модуль PFC2, то можно также указать PIR-скорость с помощью ключевого слова **arate** и параметра *peak-rate* (0 или от 32 до 8 000 000 Кбит/с).

В IOS-коммутаторе устанавливаются CIR-скорость (*rate* от 32 000 до 4 000 000 000 бит/с) и величина всплеска (`burst`, 1000–512 000 000 байтов). С помощью модуля PFC2 можно указать PIR-скорость, используя ключевое слово **pir** и параметр *peak-rate* (от 32 000 до 4 000 000 000 бит/с), а также величину всплеска (*max-burst*, 1000–512 000 000 байтов).

Совет

Заметим, что для COS- и IOS-коммутаторов используются различные единицы скорости (`rate`) и всплеска (`burst`). Для COS-коммутатора необходимо вводить скорость в килобитах в секунду и величину всплеска — в килобитах. Для IOS-коммутатора скорость задается в битах в секунду, а величина всплеска — в байтах.

В зависимости от того, как регулировщик измеряет скорость трафика, возможно применение различных действий.

- **Подтверждение передачи (confirming; in-profile-трафик — скорость меньше, чем CIR).** Трафик стандартно перенаправляется. IOS-коммутатор допускает выполнение действия `conform-action` вместо действий `drop` (фрейм отбрасывается и не перенаправляется), `policed-dscp-transmit` (внутреннее DSCP-значение понижается в процессе преобразования) и `transmit` (фрейм перенаправляется в исходном виде).
- **Превышение скорости (exceeding; непрофильный трафик — превышает CIR-скорость).** Стандартно трафик аннулируется. COS-коммутатор допускает действие `drop` (фрейм не перенаправляется) или `police-dscp` (внутреннее DSCP-значение понижается в процессе преобразования). IOS-коммутатор допускает выполнение действия `exceed-action`: `drop` (фрейм отбрасывается и не перенаправляется), `policed-dscp-transmit` (внутреннее DSCP-значение понижается в процессе преобразования) или `transmit` (фрейм перенаправляется в исходном состоянии).
- **Нарушение правил (violating; непрофильный трафик — превышает PIR-скорость).** Стандартно выполняется то же действие, что и при превышении скорости. COS-коммутатор может либо выполнить фрейм, либо применить действие `police-dscp` (внутреннее DSCP-значение понижается в процессе преобразования). IOS-коммутатор допускает применение действия `violate-action`: `drop` (фрейм отбрасывается и не перенаправляется), `policed-dscp-transmit` (внутреннее DSCP-значение понижается в процессе преобразования) или `transmit` (фрейм перенаправляется в исходном состоянии).

б) Использование регулировщика микропотока (*необязательно*).

Система COS	`set qos policer microflow microflow-name rate rate burst burst {drop | policed-dscp}`
Система IOS	См. этап 8.г, пункт 2

В COS-коммутаторе регулировщик определяется средней или CIR-скоростью (*rate*, 0 или от 32 до 8000000 Кбит/с), а также допускается превышение этой скорости — параметр *burst* (от 1 до 32000 Кб).

Совет

В коммутаторе Catalyst 6000 с платой PFC2 для любого типа регулирования микропотока необходимо включить регулирование микропотоков для хостового трафика. Соответствующие команды приведены в описании этапа 5.

В IOS-коммутаторе регулировщики микропотока конфигурируются как часть таблицы стратегий. Дополнительная информация приведена в описании этапа 8.г.

В зависимости от того, как регулировщик измеряет скорость трафика, возможно выполнение следующих действий:

- **подтверждение передачи (conforming; in-profile-трафик — скорость меньше, чем CIR)** — стандартно трафик передается дальше;
- **превышение скорости (exceeding; непрофильный трафик — превышает CIR-скорость)** — стандартно трафик аннулируется. COS-коммутатор допускает действие `drop` (фрейм не передается) или `policed-dscp` (внутреннее DSCP-значение понижается в процессе преобразования)

Совет

Указанное пользователем значение скорости (*rate*) для CIR может отличаться от фактически используемого. В аппаратном обеспечении QoS используются значения, которые образуются в результате округления указанного значения *rate* до ближайшего кратного дискретности скорости (см. табл. 13.4).

Таблица 13.4. Дискретность значений CIR-скорости

Диапазон CIR/PIR-скорости	Дискретность фактического значения
1-1 048 576 (1 Мбит/с)	32 768 (32 Кбит/с)
1 048 577-2 097 152 (2 Мбит/с)	65 536 (64 Кбит/с)
2 097 153-4 194 304 (4 Мбит/с)	131 072 (128 Кбит/с)
4 194 305-8 388 608 (8 Мбит/с)	262 144 (256 Кбит/с)
8 388 609-1 677 210 (16 Мбит/с)	524 288 (512 Кбит/с)
1 677 217-33 554 432 (32 Мбит/с)	1 048 576 (1 Мбит/с)
33 554 433-67 108 864 (64 Мбит/с)	2 097 152 (2 Мбит/с)
67 108 865-134 217 728 (128 Мбит/с)	4 194 304 (4 Мбит/с)
134 217 729-268 435 456 (256 Мбит/с)	8 388 608 (8 Мбит/с)
268 435 457-536 870 912 (512 Мбит/с)	1 677 216 (16 Мбит/с)

Окончание табл. 13.4

Диапазон CIR/PIR-скорости	Дискретность фактического значения
536 870 913-1 073 741 824 (1 Гбит/с)	33 554 432 (32 Мбит/с)
1 073 741 825-2 147 483 648 (2 Гбит/с)	67 108 864 (64 Мбит/с)
2 147 483 649-4 294 967 296 (4 Гбит/с)	134 217 728 (128 Мбит/с)
4 294 967 297-8 589 934 592 (8 Гбит/с)	268 435 456 (256 Мбит/с)

На практике рекомендуется устанавливать величину всплеска равной 32 Кбит или выше. Так как величина параметра *burst* управляет маркерной ячейкой, при выборе значения следует соблюдать осторожность. Пакеты, после прибытия которых маркерная ячейка переполнена и не поглощает всплеск, потенциально могут быть отброшены.

Таким образом, следует выбирать значение параметра *burst*, которое будет выше ожидаемого пика, делённого на 4000, а также будет превышать ожидаемый размер пакета большего принимаемого фрейма. При выборе слишком малого значения *burst* фреймы, имеющие размер больший величины всплеска, будут рассматриваться как непрофильные и удаляться. Кроме того, следует заметить, что в аппаратном обеспечении QoS используются значения, являющиеся результатом округления заданной величины *burst* до ближайшего фиксированного значения всплеска (см. табл. 13.5).

Таблица 13.5. Фиксированные значения CIR-всплесков

Диапазон CIR/PIR-всплеска	Фактическое фиксированное значение
1-32 768 (32 Кб)	1024 (1 Кб)
32 769-65 536 (64 Кб)	2048 (2 Кб)
65 537-131 072 (128 Кб)	4096 (4 Кб)
131 073-262 144 (256 Кб)	8 192 (8 Кб)
262 145-524 288 (512 Кб)	16 384 (16 Кб)
524 289-1 048 575 (1 Мб)	32 768 (32 Кб)
1 048 577-2 097 152 (2 Мб)	65 536 (64 Кб)
2 097 153-4 194 304 (4 Мб)	131 072 (128 Кб)
4 194 305-8 388 608 (8 Мб)	262 144 (256 Кб)
8 388 609-16 777 216 (16 Мб)	524 288 (512 Кб)
16 777 217-33 554 432 (32 Мб)	1 048 576 (1 Мб)
33 554 433-67 108 864 (64 Мб)	2 097 152 (2 Мб)
67 108 865-134 217 728 (128 Мб)	4 194 304 (4 Мб)
134 217 729-268 435 456 (256 Мб)	8 388 608 (8 Мб)
268 435 457-536 870 912 (512 Мб)	16 777 216 (16 Мб)

в) Включение регулирования потока для мостового трафика (*необязательно, только для коммутаторов третьего уровня*)

Система COS	`set qos bridged-microflow-policing {enable | disable} vlan-list`

Система IOS	`mls qos bridged` (режим конфигурирования интерфейса)

Регулирование микропотока обычно разрешено только для трафика, коммутируемого на третьем уровне, или для трафика, который коммутируется между VLAN-сетями. Однако можно использовать регулировщики микропотоков для мостового трафика (во внутренних сетях VLAN) в определенных VLAN-сетях. Необходимо задать список VLAN-сетей (`vlan-list` для COS-коммутатора) или использовать эту команду на VLAN-интерфейсах.

Совет

Эту команду для осуществления любого типа регулирования микропотоков нужно использовать в модуле PFC2.

г) Конфигурирование понижающего DSCP-преобразования *(необязательно)*.

Система COS	`set qos policed-dscp-map [normal | excess] `*`internal-dscp:policed-dscp...`*
Система IOS	`mls qos map policed-dscp `*`internal-dscp`*` to `*`policed-dscp`* (режим глобальной конфигурации)

Внутренние DSCP-значения (*internal-dscp*) преобразовываются в значения *policed-dscp*. Внутренние DSCP-значения можно указывать как отдельные значения, множество значений, разделенных запятыми, или в виде диапазона, заданного с использованием дефиса. В команде для COS-коммутатора между внутренним и преобразованным DSCP-значениями необходимо ввести двоеточие (:), а в команде для IOS-коммутатора — ключевое слово `to`. Чтобы задать несколько различных преобразований в COS-коммутаторе, необходимо разделять их пробелами. Для этой же цели в IOS-коммутаторе следует повторить эту команду.

Модуль PFC2 допускает понижающие преобразования DSCP-значений для регулируемого трафика в режиме `normal` (непрофильный, превышающий CIR-скорость) или `excess` (нарушение, выше PIR-скорости).

6. Определение соответствия трафика QoS-стратегии *(только для коммутаторов третьего уровня)*.

Внимание!

В COS-коммутаторе фактическая QoS-стратегия определяется путем группировки команд `set qos acl`, имеющих общие ACL-имена. В IOS-коммутаторах в первую очередь определяются списки доступа, а QoS-стратегии определяются отдельно.

Совет

В последующих этапах адреса отправителя и получателя задаются параметрами *source-ip* и *destination-ip* с масками для инверсного соответствия (0-бит соответствует, 1-бит — инвертированная маска). Если соответствующими будут любые адреса, то можно заменить поля адреса и маски ключевым словом `any`. Если совпадать должен адрес определенного узла, то можно заменить эти поля ключевым словом `host`, за которым следует IP-адрес узла.

В COS-коммутаторе списки доступа называются записями контроля доступа (*Access Control Entries — ACE*) и во время ввода содержатся в буфере редактирования. Каждой ACE-записи присваивается индексный номер. Относительным расположением записи во время ввода можно управлять посредством ключевого слова `before` (запись помещается перед записью с номером *editbuffer-index*) или `modify` (заменяет ACE-запись в позиции *editbuffer-index*).

На данном этапе COS-коммутаторы также позволяют закрепить агрегированные или микропотоковые регулировщики за ACE-записью. В IOS-коммутаторах регулировщики назначаются QoS-стратегиям в процессе выполнения этапа 5.

В COS-коммутаторе QoS-значения фрейма или пакета также могут быть заданы списком доступа. Для этого используется одно из ключевых слов: `dscp` (устанавливается DSCP-значение, равное *dscp*), `trust-cos` (устанавливается DSCP-значение, полученное из CoS-значения фрейма), `trust-ipprec` (устанавливается DSCP-значение, полученное из значения IP-приоритета пакета) или `trust-dscp` (используется существующее DSCP-значение).

Для проверки значения IP-приоритета, заданного с помощью номера (0-7) или текстовой строки, можно использовать ключевое слово `precedence`. Доступными значениями являются: `critical` (5), `flash` (3), `flash-override` (4), `immediate` (2), `internet` (6), `network` (7), `priority` (1) и `routine` (0).

Для проверки соответствия DSCP-битов, содержащихся в DS-байте IP-пакета, можно использовать ключевое слово `dscp-field` (или `dscp` в IOS). *dscp*-значение может быть задано с помощью номера (6 битов: от 0 до 63) или имени в виде текстовой строки. Доступными именами являются: `default` (000000), `ef` (101110), (гарантированное перенаправление — Assured Forwarding, AF) `af11` (001010), `af12` (001100), `af13` (001110), `af21` (010010), `af22` (010100), `af23` (010110), `af31` (011010), `af32` (011100), `af33` (011110), `af41` (100010), `af42` (100100), `af43` (100110), (селектор класса — Class Selector, CS) `cs1` (приоритет 1, 001000), `cs2` (приоритет 2, 010000), `cs3` (приоритет 3, 011000), `cs4` (приоритет 4, 100000), `cs5` (приоритет 5, 101000), `cs6` (приоритет 6, 110000) и `cs7` (приоритет 7, 111000).

IOS-коммутатор также допускает использование ключевого слова `tos` для проверки ToS-уровня (от 0 до 15). Доступными значениями являются: `max-reliability max-throughput min-delay min-monetary-cost` и `normal`.

а) Проверка соответствия адреса отправителя IP-трафика (*необязательно*).

Система COS	`set qos acl ip` *acl-name* {`dscp` *dscp* | `trust-cos` | `trust-ipprec` | `trust-dscp`} [`microflow` *microflow-name*] [`aggregate` *aggregate-name*] *source-ip source-mask* [`precedence` *precedence* | `dscp-field` *dscp*] [`before` *editbuffer-index* | `modify` *editbuffer-index*]
Система IOS	`access-list` *acc-list-number* {`permit` | `deny`} `ip` *source ip source-mask* (режим глобальной конфигурации) или `ip access-list standard` *acl-name* {`permit` | `deny`} *source-ip source-mask*}

На список доступа ссылаются по имени (параметр *acl-name*, текстовая строка) или по номеру (параметр *acc-list-number*, от 1 до 99 или от 1300 до 1999).

б) Проверка отправителя, получателя IP-трафика и номера порта (*необязательно*)

Система COS	**set qos acl ip** *acc-list* **[dscp** *dscp* **trust-cos trust-ipprec | trust-dscp] [microflow** *microflow name***] [aggregate** *aggregate-name***] {[tcp | udp] *source ip source mask [operator [source port]] destination ip destination mask [operator [dest-port]] [established] [precedence precedence | dscp-field dscp] [before** *editbuffer-index* **modify** *editbuffer-index***]**
Система IOS	**access-list** *acc-list* **{permit | deny}** *protocol source-ip source-mask [operator [source-port]] destination-ip destination- mask [operator [dest-port]] [precedence** *precedence***] [dscp** *dscp***] [tos** *tos***]** (режим глобальной конфигурации) или **ip access-list extended** *acl-name* **{permit | deny}** *protocol source-ip source-mask [operator [source-port]] destination-ip destination-mask [operator [dest-port]]* **[precedence** *precedence***] [dscp** *dscp***] [tos** *tos***]**

На список доступа ссылаются по имени (параметр *acl-name*, текстовая строка) или номеру (параметр *acc-list number*, от 100 до 199 или от 2000 до 2699)

В команде может указываться IP-протокол (*protocol*). Допускается использование следующих ключевых слов: **ip** (любой IP протокол), **tcp**, **udp**, **eigrp** (протокол маршрутизации EIGRP), **gre** (общая инкапсуляция для маршрутизации — Generic Routing Encapsulation), **icmp** (протокол управляющих сообщений Internet — Internet Control Message Protocol), **igmp** (межсетевой протокол управления группами — Internet Group Management Protocol), **igrp** (протокол маршрутизации IGRP), **ipinip** (туннель IP в IP), **nos**, **ospf** (протокол маршрутизации OSPF), номера IP-протокола (0-255).

Для указания того, каким образом сравниваются номера портов отправителя и получателя, используется ключевое слово **operator**. Можно использовать следующие операторы: **lt** (меньше чем), **gt** (больше чем), **eq** (равно), **neq** (не равно) или **range** (внутри диапазона, заданного двумя значениями номеров портов). Порты отправителя и получателя задаются номерами (0-65535) либо именами (текстовые строки).

Допустимыми именами протоколов стека TCP являются: **bgp**, **chargen**, **daytime**, **discard**, **domain**, **echo**, **finger**, **ftp**, **ftp-data**, **gopher**, **hostname**, **irc**, **klogin**, **kshell**, **lpd**, **nntp**, **pop2**, **pop3**, **smtp**, **sunrpc**, **syslog**, **tacacs-ds**, **talk**, **telnet**, **time**, **uucp**, **whois** и **www**. В дополнение к этому можно использовать ключевое слово **established** для сравнения пакетов из установленных соединений либо пакетов, имеющих установленные биты RST или ACK.

Допустимыми именами протоколов стека UDP являются: biff, bootpc, bootps, discard, dns, dnsix, echo, mobile-ip, nameserver, netbios-dgm, netbios-ns, ntp, rip, snmp, snmptrap, sunrpc, syslog, tacacs-ds, talk, tftp, time, who и xdmcp.

в) Проверка ICMP-трафика (необязательно).

Система COS	**set qos acl ip** *acl-name* {**dscp** *dscp* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow-name*] [**aggregate** *aggregate-name*] **icmp** *source-ip source-mask destination-ip destination-mask* {*icmp-type* [*icmp-code*] | *icmp-message*} [**precedence** *precedence* | **dscp-field** *dscp*] [**before** *editbuffer-index* **modify** *editbuffer-index*]
Система IOS	**access-list** *acc-list* [**permit** | **deny**] **icmp** *source-ip source-mask destination-ip destination-mask* [*icmp-type* [*icmp-code*] | *icmp-message*] [**precedence** *precedence*] [**dscp** *dscp*] [**tos** *tos*] (режим глобальной конфигурации) или **ip access-list extended** *acl-name* {**permit** | **deny**} **icmp** *source-ip source-mask destination-ip destination-mask* [*icmp-type* [*icmp-code*] | *icmp-message*] [**precedence** *precedence*] [**dscp** *dscp*] [**tos** *tos*]

На список доступа ссылаются по имени (параметр *acl-name*, текстовая строка) или номеру (параметр *acc-list-number*, от 100 до 199 или от 2000 до 2699).

В командную строку можно вместе один или несколько параметров *icmp-type*, *icmp-type*, *icmp-code* или *icmp-message*. В поле *icmp-type* указывается тип ICMP-сообщения (0–15), а в поле *icmp-code* — необязательный код ICMP-сообщения (0–255). Поле *icmp-message* — имя (текстовая строка), выбирается одно из следующих значений: **administratively-prohibited**, **alternate-address**, **conversion-error**, **dod-host-prohibited**, **dod-net-prohibited**, **echo**, **echo-reply**, **general-parameter-problem**, **host-isolated**, **host-precedence-unreachable**, **host-redirect**, **host-tos-redirect**, **host-tos-unreachable**, **host-unknown**, **host-unreachable**, **information-reply**, **information-request**, **mask-reply**, **mask-request**, **mobile-redirect**, **net-redirect**, **net-tos-redirect**, **net-tos-unreachable**, **net-unreachable**, **network-unknown**, **no-room-for-option**, **option-missing**, **packet-too-big**, **parameter-problem**, **port-unreachable**, **precedence-unreachable**, **protocol-unreachable**, **reassembly-timeout**, **redirect**, **router-advertisement**, **router-solicitation**, **source-quench**, **source-route-failed**, **time-exceeded**, **timestamp-reply**, **timestamp-request**, **traceroute**, **ttl-exceeded** или **unreachable**.

г) Проверка ICMP-трафика (*необязательно*)

Система COS	**set qos acl ip** *acl-name* {**dscp** *dscp* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow-name*] [**aggregate** *aggregate-name*] **icmp** *source-ip source-mask destination-ip destination-mask* [*icmp-type*] [**precedence** *precedence* | **dscp-field** *dscp*] [**before** *editbuffer-index* | **modify** *edit-buffer-index*]
Система IOS	**access-list** *acc-list* {**permit** | **deny**} **icmp** *source-ip source-mask destination-ip destination-mask* [*icmp-type*] [**precedence** *precedence*] [**dscp** *dscp*] [**tos** *tos*] (режим глобальной конфигурации) или **ip access-list extended** *acl-name* {**permit** | **deny**} **icmp** *source-ip source-mask destination-ip destination-mask* [*icmp-type*] [**precedence** *precedence*] [**dscp** *dscp*] [**tos** *tos*]

На список доступа ссылаются по имени (параметр *acl-name*, текстовая строка) или номеру (параметр *acc-list-number*, от 100 до 199 или от 2000 до 2699).

Если в качестве протокола задан **icmp**, то для последующей фильтрации можно добавить дополнительное поле ICMP-сообщения: **dvmrp**, **host-query**, **host-report**, **pim** или **trace**.

д) Проверка IPX-трафика (*необязательно*).

Система COS	**set qos acl ipx** *acl-name* {**dscp** *dscp* | **trust-cos**} [**aggregate** *aggregate-name*] *protocol source network* [*dest-net dest-node*] [*dest-net mask*] [*dest-node mask*] [**before** *editbuffer-index* | **modify** *editbuffer_index*]
Система IOS	**access-list** *access-list-number* {**deny** | **permit**} *source-network*[*.source node*[*source node mask*]] [*destination-network*[*.destination-node* [*destination-node-mask*]]] (режим глобальной конфигурации) или **ipx access-list standard** *acl-name* {**permit** | **deny**} *source-network*[*.source-node* [*source-node-mask*]] [*destination-network*[*.destination-node* [*destination-node-mask*]]]

На список доступа ссылаются по имени (параметр *acl-name*, текстовая строка) или номеру (параметр *acc-list-number*, от 800 до 899). Адреса определяются параметрами *source-network* и *destination-network* (восьмизначное шестнадцатеричное число, от 1 до FFFFFFFE; -1 обозначает всю сеть), параметрами *source-node* и *destination-node* (48-битовый MAC-адрес в трехбайтовом

формате с разделенными точками), а также параметрами *source-node-mask* и *destination-node-mask* (48-битовые маски в трехбайтном формате с разделением точками, бит 1 игнорируется или действует как инвертированная маска).

ж) Проверка трафика MAC-уровня (*необязательно*).

Система COS	**set qos acl mac** *acl-name* {**dscp** *dscp* | **trust-cos**} [**aggregate** *aggregate name*] [{*source mac source-mask* | **any**} {*dest-mac dest-mask* | **any**} [*ether-type*] [**before** *editbuffer-index* | **modify** *editbuffer index*]
Система IOS	**mac access-list extended** *acl-name* (режим глобальной конфигурации) {**permit** | **deny**} {*source-mac source-mask* | **any**} {*dest-mac dest-mask* | **any**} *ether-type*

На список доступа ссылаются по имени *acl-name* (текстовая строка).

Для проверки указываются MAC-адреса отправителя и получателя (source и destination), а также маски (source-mask и destination-mask). Адресами являются 48-битовые MAC-адреса, записанные в виде трех разделенных точками групп, состоящих из четырех шестнадцатеричных чисел (т.е. 0000.1111.2222). Бит 1 в маске приводит к тому, что бит адреса будет проигнорирован.

В COS-коммутаторах поле *ether-type* может принимать одно из следующих значений: **ethertalk** (0x809b), **aarp** (0x8053), **dec-mop-dump** (0x6001), **dec-mop-remote-console** (0x6002), **dec-phase-iv** (0x6003), **dec-lat** (0x6004), **dec-diagnostic-protocol** (0x6005), **dec-lavc- sca** (0x6007), **dec-amber** (0x6008), **dec-mumps** (0x6009), **dec-lanbridge** (0x8038), **dec-dsm** (0x8039), **dec-netbios** (0x8040), **dec-msdos** (0x8041), **banyan-vines-echo** (0x0bad), **xerox-ns-idp** (0x0600) или **xerox-address-translation** (0x0601).

В IOS коммутаторах поле *ether type* может принимать одно из следующих значений: **aarp** (0x8053), **amber** (0x6008), **appletalk** (0x809b), **diagnostic** (0x6005), **dec-spanning** (0x8038), **dsm** (0x8039), **etype-6000** (0x6000), **etype-8042** (0x8042), **lat** (0x6004), **lavc-sca** (0x6007), **mop-console** (0x6002), **mop-dump** (0X6001), **msdos** (0X8041), **mumps** (0x6009), **netbios** (0x8040), **vines-ip** (0x0bad), **vines-echo** (0x0baf) или **xns-idp** (0x0600).

з) Создание стандартных условий проверки (*необязательно, только для COS-коммутаторов*).

Совет

При первом включении QoS-функций стандартным действием является негарантированная ("best effort") доставка трафика. Это связано с тем, что внутренние DSCP- и CoS-значения равны нулю. Регулирование не конфигурируется и не осуществляется.

- Установка стандартного действия для IP-пакетов (*необязательно*).

  Если другой соответствующей записи IP ACE для фрейма не существует, то внутреннее DSCP-значение устанавливается согласно использованному ключевому слову: **dscp** (DSCP устанавливается равным параметру *dscp*), **trust-cos**

(DSCP преобразовывается из входного CoS), `trust-ipprec` (DSCP преобразовывается из входящего значения IP-приоритета) или `trust-dscp` (используется входящее значение DSCP). Стандартные регуляторщики могут быть заданы с помощью ключевых слов `microflow` и `aggregate`.

Система COS	`set qos acl default-action ip {dscp dscp | trust-cos | trust-ipprec | trust-dscp} [microflow microflow-name] [aggregate aggregate-name]`
Система IOS	Нет

- Установка стандартного действия для IPX-пакетов (*пропускающего*).

Система COS	`set qos acl default-action ipx {dscp dscp | trust-cos} [microflow microflow-name] [aggregate aggregate name]`
Система IOS	Нет

Если другой соответствующей записи IPX ACE для фрейма не существует, то внутреннее DSCP-значение устанавливается согласно использованному ключевому слову: `dscp` (DSCP устанавливается равным параметру *dscp*) или `trust-cos` (DSCP преобразовывается из входящего CoS). Стандартные регуляторщики могут быть заданы с помощью ключевых слов `microflow` и `aggregate`.

- Установка стандартного действия для MAC-трафика (*пропускающего*).

Система COS	`set qos acl default-action mac {dscp dscp | trust-cos} [microflow microflow-name] [aggregate aggregate-name]`
Система IOS	Нет

Если другой соответствующей записи MAC ACE для фрейма не существует, то внутреннее DSCP-значение устанавливается согласно использованному ключевому слову: `dscp` (DSCP устанавливается равным параметру *dscp*) или `trust-cos` (DSCP преобразовывается из входящего CoS). Стандартные регуляторщики могут быть заданы с помощью ключевых слов `microflow` и `aggregate`.

7. Группировка соответствующего трафика в таблицу класса (*только для IOS-коммутаторов*).

   a) Создание таблицы класса (class map).

Система COS	Нет
Система IOS	`class-map class name [match-all | match-any]` (режим глобальной конфигурации)

Для таблицы класса с именем, указанным параметром *class-name* (текстовая строка), определяется одно или несколько условий проверки. Можно осуществлять проверку по всем условиям (`match-all`, стандартная установка) или по любому из них (`match-any`).

б) Использование списка доступа для трафика с предполагаемым совпадением (*необязательно*)

Система COS	Нет
Система IOS	`match access-group name` *acc-list*

С помощью таблицы класса проверяется трафик, который разрешен списком доступа с именем, заданным параметром *acc-list* (именованным или пронумерованным). Список доступа конфигурируется на этапе 6.

в) Проверка соответствия значениям IP-приоритета (*необязательно*)

Система COS	Нет
Система IOS	`match ip precedence` *ipprec1 [...ipprec8]*

Для проверки может быть задано до восьми значений IP-приоритета (*ipprec*: от 0 до 7), разделенных пробелами. Доступными значениями являются: **critical** (5), **flash** (3), **flash-override** (4), **immediate** (2), **internet** (6), **network** (7), **priority** (1) и **routine** (0).

г) Проверка соответствия DSCP-значениям (*необязательно*).

Система COS	Нет
Система IOS	`match ip dscp` *dscp1 [...dscp8]*

Для проверки может быть задано до восьми DSCP-значений, разделенных пробелами. DSCP-значения можно задавать с помощью номера (6 битов, от 0 до 63) или по имени (текстовая строка). Доступными именами являются: **default** (000000), **ef** (экспресс-коммутация — Express Forwarding, EF, 101110), (гарантированная пересылка — Assured Forwarding, AF) **af11** (001010), **af12** (001100), **af13** (001110), **af21** (010010), **af22** (010100), **af23** (010110), **af31** (011010), **af32** (011100), **af33** (011110), **af41** (100010), **af42** (100100), **af43** (100110), (селектор класса — Class Selector, CS) **cs1** (приоритет 1, 001000), **cs2** (приоритет 2, 010000), **cs3** (приоритет 3, 011000), **cs4** (приоритет 4, 100000), **cs5** (приоритет 5, 101000), **cs6** (приоритет 6, 110000) и **cs7** (приоритет 7, 111000).

8. Определение QoS-правила *(только для IOS-коммутаторов)*.

а) Создание правила.

Система COS	Нет
Система IOS	`policy-map` *policy-name* (режим глобальной конфигурации)

б) Использование одной или нескольких таблиц класса для поиска соответствующего трафика.

- Использование существующей таблицы класса (*необязательно*)

Система COS	Нет
Система IOS	`class` *class-name*

Если таблица класса уже определена, то на нее может быть дана ссылка по имени (*class-name*, текстовая строка).

- Создание новой таблицы класса (*необязательно*).

Система COS	Нет
Система IOS	`class` *class-name* [`access-group` *acc-list* | `dscp` *dscpi* [... *dscpN*] `precedence` *ipprec1* [... *ipprecN*]]

Таблица класса также может быть создана в процессе конфигурирования правил. В этом случае предоставляется более эффективный способ определения таблиц классов.

в) Установка состояния QoS-доверия (*необязательно*).

Система COS	Нет
Система IOS	`trust` {`cos` | `dscp` | `ip-precedence`}

IOS-коммутаторы способны избирательно определять отправителей для внутренних DSCP-значений из входящего трафика. Для фреймов, соответствующих таблице класса, возможны несколько вариантов получения DSCP-значения. `cos` (используется CoS DSCP-преобразование), `dscp` (используется входящее DSCP в исходном виде) или `ip-precedence` (используется ToS DSCP-преобразование).

г) Использование регулировщика для управления полосой пропускания соответствующего трафика.

- Использование именованного агрегированного регулировщика (*необязательно*).

Система COS	Нет
Система IOS	`police aggregate` *policer-name*

Регулировщик с именем, определенным в параметре *policer-name* (текстовая строка), управляет агрегированным трафиком от всех входных портов, за которыми он закреплен.

- Указание интерфейсного регулировщика для контроля одного интерфейса (*необязательно*).

Система COS	Нет
Система IOS	`police` [`aggregate` *policer-name*] [`flow`] *rate burst* [*max-burst*] [`pir` *peak-rate*] [`conform-action` *action*] [`exceed-action` *action*] [`violate-action` *action*]

Если регулировщик определен как часть правила, то он управляет только агрегированным трафиком от входных портов, на которых назначена данная стратегия. Для определения агрегированного регулировщика используется ключевое слово `aggregate`, а для определения регулировщика по микропотокам — ключевое слово `flow`.

Коммутатор Catalyst 3550 позволяет указывать только действие `exceed-action`.

Совет

Чтобы использовать регулировщики микропотоков в IOS-коммутаторе, необходимо сначала включить функции микропотоков с помощью команды `mls qos flow-policing`. В дополнение к этому также должна быть включена микропотоковая регулировка мостового трафика в модуле PFC2, или же следует использовать регулировку маршрутизируемого трафика. Микропотоковая регулировка мостового трафика включается с помощью команды `mls qos bridged` на VLAN-интерфейсе.

Необходимо также установить CIR-скорость (параметр *rate*, от 32 000 до 4 000 000 000 бит/с) и размер всплеска (*burst*, от 1000 до 512 000 000 байтов). При использовании модуля PFC2 также можно указать PIR-скорость с помощью ключевого слова `pir` и параметра *peak-rate* (от 32 000 до 4 000 000 000 бит/с), а также максимальный размер всплеска *max-burst* (от 1000 до 512 000 000 байтов).

Совет

Значение *rate*, задаваемое пользователем для CIR или PIR-скорости, может отличаться от фактически используемого. Диапазоны *rate* и *burst*, а также фактические значения дискретности приведены в описании этапа 5.б.

На практике рекомендуется устанавливать величину всплеска равной 32 Кб (32 КБ — для COS, 4096 байтов — для IOS) или выше. Т.к. величина параметра *burst* управляет маркерной ячейкой, при выборе значения следует соблюдать осторожность. Пакеты, в результате прибытия которых маркерная ячейка превышает величину всплеска, потенциально могут быть отброшены.

Таким образом, следует выбирать такое значение параметра *burst*, которое будет выше значения *rate*, делённого на 4000, а также будет превышать ожидаемый размер наибольшего принимаемого фрейма. При выборе слишком малого значения *burst* фреймы, имеющие размер, больший величины всплеска, будут рассматриваться как непрофильные и удалиться.

В зависимости от того, как регулировщик измеряет скорость трафика, возможно применение различных действий.

- **Подтверждение скорости** (conforming; **ip-profile-трафик — скорость меньше, чем CIR**). Трафик стандартно перенаправляется дальше. IOS-коммутатор допускает выполнение действия `conform-action` вместо действий `drop` (фрейм отбрасывается и не перенаправляется), `set-dscp-transmit new-dscp` (DSCP-значение устанавливается равным *new dscp*), `set-prec transmit new precedence` (IP-приоритет устанавливается равным *new-precedence*) и `transmit` (фрейм перенаправляется в исходном виде).
- **Превышение скорости** (exceeding; **непрофильный трафик — превышает CIR-скорость**). Стандартно трафик отбрасывается. IOS-коммутатор допускает выполнение действия `exceed-action`: `drop` (фрейм отбрасывается и не перенаправляется), `policed-dscp-transmit` (внутреннее DSCP-значение понижается в процессе преобразования) или `transmit` (фрейм перенаправляется в исходном состоянии).
- **Нарушение** (violating; **непрофильный трафик — превышает PIR-скорость**). Стандартно предпринимается то же действие, что и при превышении скорости. IOS-коммутатор допускает выполнение следующих действий `violate-action`: `drop` (фрейм отбрасывается и не перенаправляется), `policed-dscp-transmit` (внутреннее DSCP-

значение понижается в процессе преобразования) и `dynamic` (фрейм перенаправляется в исходном состоянии).

9. Применение правила к порту коммутатора или VLAN-сети.

   а) Завершение ACE-записи правила (*только для коммутаторов третьего уровня*).

Система COS	`commit qos acl {acl-name  all}`
Система IOS	Нет

Записи ACE в процессе ввода содержатся во временном буфере редактирования. После редактирования ACL необходимо подтвердить (т. е. сохранить) в NVRAM-память либо определенную ACE-запись (параметр *acl-name* — текстовая строка), либо все записи (`all`). Кроме того, подтвержденные ACE-записи компилируются и записываются в аппаратное обеспечение коммутаторов для использования в дальнейшем.

Совет

Если ACE-запись была предварительно прикреплена к порту коммутатора, то любые изменения в ней вступают в действие немедленно при ее повторном подтверждении. Для удаления неподтвержденной ACE-записи из буфера редактирования используется команда `rollback qos acl {acl-name | all}`.

Для внесения изменений в подтвержденную ACE-запись необходимо с помощью команды `show qos acl info` *acl-name* определить ее индекс в списке доступа. Затем с помощью команды `clear qos acl` *acl-name index* удалить ACE-запись в позиции *index*. Добавить при необходимости дополнительные ACE и перекомпилировать список доступа с помощью команды `commit qos acl` на данном этапе.

   б) Подключение правила к порту.

Система COS	`set qos acl map` *acl-name {mod/port | vlan}*
Система IOS	`service-policy input` *policy-name* (режим конфигурирования интерфейса)

Правило (в системе COS — параметр *acl-name*; в системе IOS — параметр *policy-name*) с помощью этой команды закрепляется за портом коммутатора для немедленного применения к входящему трафику. Если стратегия будет использоваться для VLAN QoS-функций, то она закрепляется за VLAN-сетью номер *vlan* или VLAN-интерфейсом.

10. Точная настройка очередей порта выхода (*необязательно*)

    а) Настройка соотношения исходящей очереди (*необязательно*)

Система COS	`set qos txq-ratio` *port-type queue1 queue2 [queue3] queue-priority*
Система IOS	`wrr-queue queue-limit` *queue1 queue2 [queue3] queue-priority* (режим конфигурирования интерфейса)

Совет

Стандартно в портах выхода используется предотвращение перегрузок и механизм планирования, которые приведены в табл. 13.6.

Таблица 13.6. Планирование очередей и пороговые значения функции предотвращения перегрузок

Тип очереди	Пороги COS: процент отбрасывания последних пакетов, или нижний /верхний порог WRED (%)					
	Стандартная очередь 1		Стандартная очередь 2		Стандартная очередь 3	
	T1	T2	T1	T2	T1	T2
2q2t	0,1:80	2,3:100	4,5:80	6,7:100		
1p2q2t	0,1:50	2,3:60	4 80	6,7:100		
1p3q1t	0,1:100		2,3,4:100		6,7:100	
1p2q1t	0,1,2,3 70/100		4,6,7: 70/100			

Порты всех типов устанавливают фреймы с CoS-значением 5 в свои очереди строгого приоритета (кроме порта **2q2t**, не имеющего такой)

Внимание!

При использовании этой команды все порты проходят цикл отключения-включения каналов.

Необходимо оценить соотношение обычного трафика (низкого и высокого приоритета) и трафика строгого приоритета к общему количеству трафика, выходящему из порта коммутатора. Для установки процентных частей (от 1 до 100) для стандартных очередей передачи используются значения *queue1*, *queue2* и *queue3*. Чтобы установить процентную часть (от 1 до 100 процентов) очереди строгого приоритета, используется значение *queue-priority*. В сумме эти значения должны составлять 100 процентов.

В табл. 13.7 перечислены стандартные разделения буферов порта коммутатора.

Таблица 13.7. Стандартные разделения буфера порта коммутатора

Тип порта	Низкий приоритет (%)	Средний приоритет (%)	Высокий приоритет (%)	Строгий приоритет (%)
2q2t	80 (очередь 1)	Нет	20 (очередь 2)	Нет
1p2q2t	70 (очередь 1)	Нет	15 (очередь 2)	15 (очередь 3)
1p3q1t	25 (очередь 1)	25 (очередь 2)	25 (очередь 3)	25 (очередь 4)
1p2q1t	50 (очередь 1)	Нет	30 (очередь 3)	20 (очередь 3)

б) Регулировка веса обслуживания очереди передачи (необязательно)

Система COS	`set qos wrr port-type weight1 weight2 [weight3]`
Система IOS	`wrr-queue bandwidth weight1 weight2 [weight3]` (режим конфигурирования интерфейса)

Стандартные очереди портов типов (`port-type`) `1q2t`, `1p2q2t`, `1p3q1t` и `2p2q1t` обслуживаются в режиме WRR. Очередь строгого приоритета обслуживается всегда независимо от других очередей, после чего последовательно обслуживаются все стандартные очереди согласно значениям их веса (`weight`). Вес каждой очереди определяется относительно других очередей.

Стандартно порты с двумя очередями имеют соотношение 4:255, а порты с тремя очередями — соотношение 100:150:200. (Если QoS-функции отключены, то все очереди имеют равный вес — 255.)

Совет

Значение *weight* для очереди определяет количество байтов, отправляемых до того, как будет обслуживаться следующая очередь. Всегда передается целый фрейм, даже если установлено меньшее значение. Следовательно, необходимо выбрать такое значение параметра `weight` (очередь с наименьшим приоритетом), которое по крайней мере равно величине MTU (наибольшего блока, который может быть отправлен). Затем следует пропорционально установить значения остальных весов.

Чем больший вес установлен для высокоприоритетных очередей, тем больше пройдет времени, прежде чем будут обслужены низкоприоритетные очереди, что увеличивает задержку для очередей с более низким приоритетом.

После установки веса для порта необходимо убедиться, что в аппаратном обеспечении используется соответствующее значение. Для этого необходимо ввести команду `show qos info runtime mod/port` и найти сведения о коэффициенте очереди передачи. Эта команда отображает коэффициент и количество байтов.

11. Преобразование внутренних DSCP-значений в исходящие CoS-значения *(необязательно; только для коммутаторов третьего уровня)*

Система COS	`set qos dscp-cos-map` *dscp-list*`:`*cos-value* ...
Система IOS	`mls qos map dscp-cos` *dscp-list* `to` *cos-value* (режим глобальной конфигурации)

Для каждого фрейма внутреннее DSCP-значение генерирует окончательное CoS-значение. Окончательное CoS-значение записывается в CoS-поля исходящих магистральных каналов, а также используется для управления исходящим расписанием и функцией предотвращения перегрузок.

В качестве DSCP-значений (*dscp-list*) может выступать одно значение (от 0 до 63), диапазон значений, заданный с помощью дефиса, или множество значений и диапазонов, разделенных запятыми. CoS-значение задается параметром `cos-value` (от 0 до 7).

В табл. 13.8 проиллюстрировано стандартное DSCP-CoS-преобразование.

Таблица 13.6. Стандартное DSCP-CoS-преобразование

DSCP	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63
CoS	0	1	2	3	4	5	6	7

Можно повторять IOS-команду до тех пор, пока не будут определены все необходимые CoS-преобразования.

12. Настройка исходящего расписания и предотвращения перегрузок (*необязательно*).

а) Использование стандартных очередей приема с отбрасыванием последних пакетов (*необязательно*).

Система COS	`set qos drop-threshold` *`port-type`* `tx queue` *`queue-id threshold-percent-1 threshold-percent-2`*
Система IOS	`wrr-queue threshold` *`queue-id threshold-percent-1 threshold-percent-2`* (режим конфигурирования интерфейса)

Для портов коммутатора с типом *2q2t* можно использовать стандартную функцию предотвращения перегрузок путем отбрасывания последних пакетов. Для каждого порога можно назначить процент буферного пространства, которое будет доступно для передачи фреймов. Значения *`threshold-percentage-n`* (от 1 до 100 процентов) задаются в последовательном порядке. Если размер буфера достигает порогового уровня, то новые исходящие фреймы отбрасываются. Стандартно порог 1 устанавливается на уровне 100 процентов, порог 2 — 60 процентов.

б) Использование WRED-очередей приема (*необязательно*).

Система COS	`set qos wred` *`port-type`* `tx queue` *`queue-id [thr1-min:]thr1 max [thr2-min:]thr2-max...`*
Система IOS	`wrr-queue random-detect min-threshold` *`queue-id thr1-min thr2-min ...`* `wrr-queue random-detect max-threshold` *`queue-id thr1 max thr2 max ...`* (обе команды вводятся в режиме конфигурирования интерфейса)

Для портов с типами *1p2q2t*, *1p3q1t* и *1p2q1t* используется метод WRED. Идентификатор очереди (`queue-id`) равен единице (стандартная низкоприоритетная очередь) или двум (стандартная высокоприоритетная очередь), кроме портов *1p3q1t*, в которых добавляется очередь 3 (стандартная очередь наивысшего приоритета). Для каждого из порогов очереди используются два ограничения: минимум (`thr1-min` от 1 до 100 процентов) и максимум (`thr2-max` от 1 до 100 процентов).

Если размер буфера ниже минимального уровня, фреймы не отбрасываются. По мере того как буфер превышает минимум, но не достигает максимума, вероятность отбрасывания фреймов возрастает. При превышении максимального уровня все фреймы отбрасываются.

Совет

Чтобы в очередях передачи IOS типов 1p3q1t и 1p2q1t включить пороги отбрасывания WRED, необходимо ввести команду `wrr-queue random-detect` *`queue-id`*.

в) Настройка планирования для исходящего трафика (*необязательно*).

Система COS	`set qos map` *`port-type tx queue-id threshold-id cos cos-list`*

Система IOS	`wrr-queue cos-map` *queue-id threshold-id cos-list* (режим конфигурирования интерфейса)

Исходящие фреймы с определенными CoS-значениями могут быть преобразованы и отправлены в указанные исходящие очереди и пороги. В поле *cos-list* может быть введено одно значение (от 0 до 7), множество значений, разделенных запятыми, или диапазон значений, заданный с использованием дефиса. Такое преобразование устанавливается для всех портов коммутатора (в COS коммутаторах) или по интерфейсам (IOS-коммутаторы).

Параметр *port-type* — тип доступной очереди, который можно просмотреть с помощью команды `show port capabilities` (COS) или `show queueing interface` (IOS). Параметры *queue-id* (1 для стандартной очереди или 2 — для очереди строгого приоритета) и *threshold-id* (1–4) определяют очередь и порог для размещения исходящих фреймов. Диапазон значений зависит от аппаратного обеспечения порта коммутатора.

Стандартные варианты преобразования показаны в табл. 13.6.

Пример конфигурирования QoS-функций

Коммутатор Catalyst 6000 с модулем PFC2 используется в качестве коммутатора уровня распределения или основного уровня. Уровень доступа состоит из нескольких платформ коммутации: коммутатор Catalyst 6000 с модулем PFC2 (или Catalyst 4000 с блоком Supervisor III), коммутатор Catalyst 4000 и Catalyst 2900XL или 3500XL. Диаграмма сети приведена на рис. 13.5.

В нашем примере необходимо использовать функции QoS для управления трафиком двух типов:

- однонаправленный протокол совместного использования файлов, TCP-порт 1214;
- трафик IP-телефонии.

Трафик на TCP-порту 1214 следует регулировать таким образом, чтобы коммутационная скорость через коммутатор уровня распределения не превышала уровня 1 Мбит/с. С другой стороны, трафик IP-телефонии следует маркировать так, чтобы он получал наивысший уровень качества обслуживания.

Порт 2/1 представляет собой магистральный Gigabit Ethernet-канал к коммутатору Catalyst 6000 на уровне доступа. Т.к. коммутатор уровня доступа способен изучать и маркировать весь трафик в любой VLAN-сети внутри магистрального канала, QoS-функции применяются на основе портов. Порт 3/1 представляет собой магистральный Gigabit Ethernet-канал либо к коммутатору Catalyst 4000, либо к коммутатору модели 3500XL на уровне доступа. Эти коммутаторы не обладают возможностями по изучению и маркированию трафика на третьем уровне, поэтому QoS-функции применяются в зависимости от VLAN-сети (`vlan-based`; каждая VLAN сеть рассматривается независимо). Выбор сделан в пользу коммутатора уровня доступа в целях демонстрации QoS-параметров для различных платформ коммутации.

Поскольку в рассматриваемой сети настраивается сквозное обеспечение качества обслуживания, можно с уверенностью предположить, что коммутаторы уровня доступа будут классифицировать и маркировать управляющий трафик IP-телефонии (TCP-порты 2000–2002) DSCP-значением 26 (AF31), а пакеты *транспортного протокола ре-*

альном времени (Real-Time Transport Protocol — RTP) или пакеты голосового носителя — DSCP-значением 46 (EF)

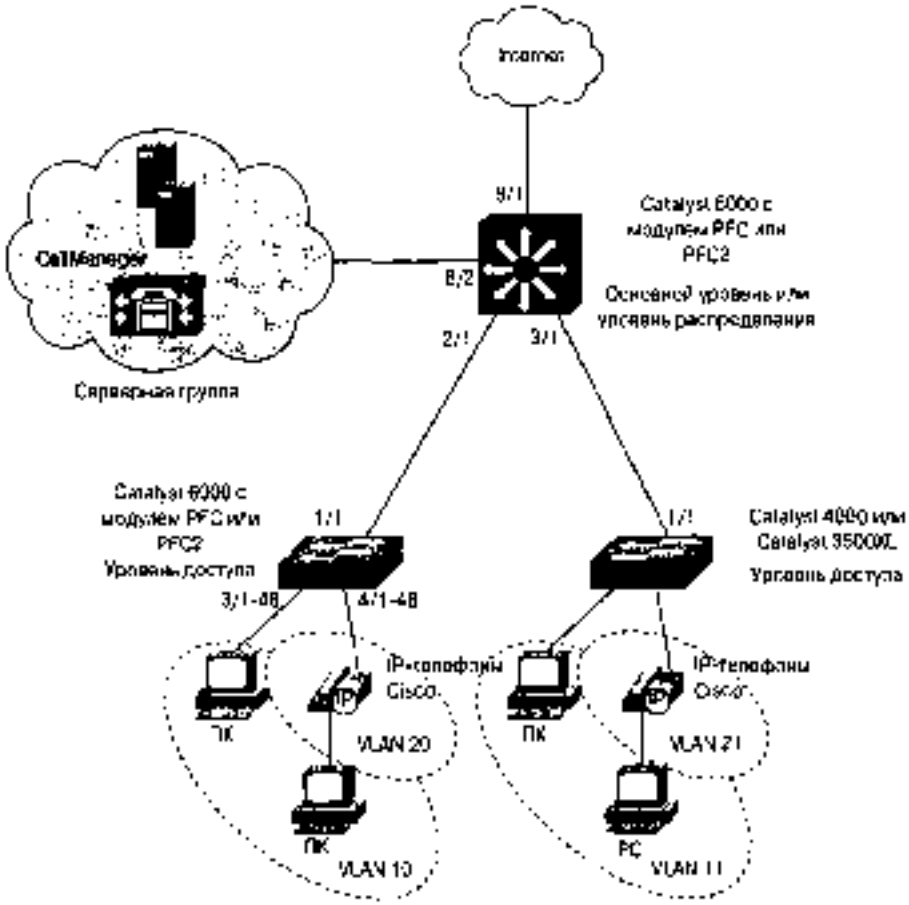


Рис. 13.5. Диаграмма сети (пример конфигурирования QoS-функций)

Порт 8/1 — Gigabit Ethernet-соединение с открытой сетью Internet, поэтому QoS-информации из этого источника не доверяют. Порт 8/2 является Gigabit Ethernet-соединением с серверной группой, в которой DSCP-информации можно доверять. Сервер IP-телефонии (например, Cisco CallManager) расположен в данной серверной группе.

Для тех входных портов, которым не доверяют, настраивается преобразование CoS-DSCP, а также преобразование IP-приоритета в DSCP-значение. Эти преобразования несколько отличаются от стандартных настроек преобразования CoS 3 в DSCP 26 (AF31) и CoS 5 в DSCP 46 (EF).

Агрегированный регулировщик потоков с именем `Policer-1214` ограничивает пропускную способность для соответствующего трафика до 1000 Кбит/с или 1 Мбит/с. Трафик, скорость которого превышает этот уровень, отбрасывается. Регулировщик применяется к трафику с TCP-портом 1214 в качестве порта отправителя или получателя. Поскольку используется протокол равноправных узлов, трафик должен соответствовать параметрам регулировщика в обоих направлениях, что позволяет передавать фай-

лы в обоих направлениях. Регулировщик и ACL-список проверки или таблица класса применяются к портам входа 2/1, 3/1 и 8/1, т.е. ко всем портам, где могут присутствовать файловые серверы.

Трафик IP-телефонии сравнивается со списком доступа в COS-коммутаторе и таблицей класса в IOS-коммутаторе. Диапазон TCP-портов 2000-2002 представляет управляющий трафик IP-телефонии между IP-телефонами и сервером CallManager. Для этого трафика устанавливается DSCP-значение 26 (AF31). Такая стратегия применяется только ко VLAN-сети 21, к которой к коммутаторам уровня доступа Catalyst 4000 и 3500XL подключены IP-телефоны. Коммутатор уровня доступа Catalyst 6000 способен осуществлять маркирование для своих IP-телефонов. IP-телефоны всегда маркируют RTP-трафик или голосовой трафик CoS-значением 5, которое преобразовывается в DSCP-значение 46 (EF) на входных портах.

Следует заметить, что в IOS-коммутаторе используется регулировщик для маркирования управляющего голосового трафика DSCP-значением 26. Этот метод является единственным, который можно использовать для маркирования DSCP-значений. Задаются фиктивные значения CIR-скорости, всплеска и PIR-значения (32000 бит/с). Поскольку трафик передается как для согласованных, так и для несогласованных значений, величины CIR и всплеска не имеют значения.

Наконец, трафик IP-телефонии регулируется расписанием в очередях портов выхода. Стандартно весь трафик с CoS-значением 5 (трафик голосового носителя) назначается очереди строгого приоритета на всех портах выхода. Для назначения фреймов с CoS-значением 3 (управляющий трафик IP-телефонии) в очередь 2 с порогом 1 определяется таблица преобразования. В результате управляющий трафик IP-телефонии размещается с опережением обычного трафика в очереди 1, но позади трафика носителя в приоритетной очереди.

Прежде всего рассмотрим конфигурацию коммутатора Catalyst 6000 уровня распределения или основного уровня.

Система COS

```
set qos enable
set port qos 2/1 port-based
set port qos 2/1 trust trust-dscp
set port qos 3/1 vlan-based
set port qos 8/1 port-based
set port qos 8/1 trust untrusted
set port qos 8/1 cos 0
set port qos 8/2 port-based
set port qos 8/2 trust trust-dscp
set qos cos-dscp-map 0 8 16 26 32 46 48 56
set qos ipprec-dscp-map 0 8 16 26 32 46 48 56

set qos policer aggregate Policer-1214 rate 1000 burst
32 drop
set qos acl ip ACL-1214 dscp 0 aggregate Policer-1214
tcp any any eq 1214
set qos acl ip ACL-1214 dscp 0 aggregate Policer-1214
tcp any eq 1214 any
commit qos acl ACL-1214
set qos acl map ACL-1214 2/1,3/1,8/2

set qos acl ip ACL-IPT-Control dscp 26 tcp any any
```

	range 2000 2002 commit qos acl ACL-IPT-Control set qos acl map ACL-IPT-Control 21 set qos map lp2q2t tx 2 1 cos 3
Система IOS	mls qos interface gig 2/1 mls qos trust dscp interface gig 3/1 mls qos vlan-based interface gig 8/1 no mls qos trust mls qos cos 0 interface gig 8/2 mls qos trust dscp mls qos map cos-dscp 0 8 16 26 32 46 48 56 mls qos aggregate-policer Policer-1214 1000000 32768 conform-action transmit exceed-action drop ip access-list extended ACL-1214 permit tcp any any eq 1214 dscp 0 permit tcp any eq 1214 any dscp 0 exit ip access-list extended ACL-IPT-Control permit tcp any any range 2000 2002 exit class-map Class-1214 match-all match access-group name ACL-1214 exit class-map Class-IPT-Control match-all match access-group 101 policy-map Policy-1214-IPT class Class-1214 police aggregate Policer-1214 class Class-IPT-Control police 32000 32000 32000 conform-action set-dscp- transmit 26 exceed-action transmit exit interface gig 2/1 service-policy input Policy-1214-IPT wrr-queue cos-map 2 1 3 interface gig 3/1 service-policy input Policy-1214-IPT wrr-queue cos-map 2 1 3 interface gig 8/1 service-policy input Policy-1214-IPT wrr-queue cos-map 2 1 3

Конфигурация коммутатора уровня доступа Catalyst 6000 показана ниже. Порт 1/1 — восходящий канал к уровню распределения или основному уровню, которым также входит в данный QoS-домен. Следовательно, DSCP-информации можно доверять. Порты 2/1-48 подключаются к персональным компьютерам конечных пользователей, поэтому QoS-информации доверять нельзя, и входящие CoS-значения понижаются до 0.

Порты 3/1-48 подключены к IP-телефонам Cisco. Входящим CoS-значениям можно доверять, поскольку они корректно обрабатываются IP-телефонами. Телефоны получают инструкции не доверять своим портам доступа и понижать поступающие с них QoS-значения 0. Эти порты также конфигурируются для осуществления QoS функций на основе VLAN-сетей, поскольку все QoS-правила будут применены к голосовой VLAN в целом.

Для классификации управляющего трафика IP-телефонии (TCP-порты 2000-2002) и маркирования внутренних DSCP-значений 26 (AF31) определен список доступа ACL-IPT-Control. Он определяется для всего трафика сети VLAN 20.

Расписание исходящей очереди конфигурируется для назначения фреймов CoS 3 (управляющий трафик IP-телефонии) в очередь 2, порог 1 во всех портах 1p2q2t (Gigabit Ethernet) и 2q2t (телефоны 10/100). Стандартно фреймы с CoS-значением 5 (трафик голосового носителя) назначаются в очередь строгого приоритета.

Система COS

```
set qos enable
set port qos 1/1 trust trust-dscp
set port qos 2/1-48 trust untrusted
set port qos 2/1-48 cos 0
set port qos 3/1-48 trust trust-cos
set port qos 3/1-48 trust-ext untrusted
set port qos 3/1-48 vlan-based
set qos cos-dscp-map 0 8 16 26 32 46 48 56
set qos ipprec-dscp-map 0 8 16 26 32 46 48 56

set qos acl ip ACL-IPT-Control dscp 26 tcp any any
range 2000 2002
commit qos acl ACL-IPT-Control
set qos acl map ACL-IPT-Control 20

set qos map 1p2q2t tx 2 1 cos 3
set qos map 2q2t tx 2 1 cos 3
```

Система IOS — Нет

Наконец, рассмотрим QoS-конфигурацию коммутаторов уровня доступа Catalyst 4000 (COS) и 3500XL (IOS). Ни одна из указанных платформ не способна классифицировать входящие фреймы. Следовательно, только коммутатор Catalyst 3500XL конфигурируется для передачи IP-телефонам Cisco, непосредственно подключенным к нему, инструкции по перезаписи CoS-значений их портов доступа в 0.

Коммутатор Catalyst 4000 должен доверять всем CoS значениям, которые отправляются конечными устройствами. Однако в расписание его очереди можно внести некоторые коррективы. Стандартно фреймы с CoS-значением 3 (например, управляющий трафик IP-телефонии) назначаются в очередь 1, порог 2, как и фреймы CoS 2. Для повышения общей эффективности обработки голосового трафика фреймы CoS 3 и CoS 5 (трафик голосового носителя) назначаются в очередь 2, порог 1.

В данном случае для правильной обработки QoS-правил должен использоваться следующий, более высокий уровень коммутаторов

Система COS	`set qos enable` `set qos map 2q1t 2 1 cos 3-5`

Система IOS	`interface fast 0/1`
	`switchport priority extend cos 0`
	`interface fast 0/2`
	`switchport priority extend cos 0`
	`interface fast 0/3 ...`

Отображение сведений о QoS-конфигурации

В табл. 13.9 приводятся команды коммутатора, которые можно использовать для получения полезной информации о конфигурации и функциях обеспечения качества обслуживания.

Таблица 13.9. Команды для отображения QoS-конфигурации и информации о функциях

Функция отображения	Операционная система коммутатора	Команда
QoS-параметры порта	COS	`show port qos` *[mod[/port]]*
	IOS	`show mls qos` {*type number* | `port-channel` *number* | `vlan` *vlan-id*}
Расписание порта и функция предотвращения перегрузок	COS	`show port qos` *[mod[/port]]* или `show qos info` {`runtime` | `config`} *{mod/port}* или `show qos info config port_type` {`tx` | `rx`}
	IOS	`show queueing interface` {*type number* | `Null` *interface number* | `vlan` *vlan id*}
QoS-преобразование	COS	`show qos maps` {`config` | `runtime`} [{`cos-dscp-map` | `ipprec-dscp-map` | `dscp-cos map` | `policed-dscp-map` {`normal-rate` | `excess-rate`}}]
	IOS	`show mls qos maps`
Регуляризаторы	COS	`show qos policer` {`config` | `runtime`} {`microflow` *[policer-name]* | `aggregate` *[policer-name]* | `all`}
	IOS	`show mls qos aggregate policer` *[aggregate-name]*

Окончание табл. 13.9

Функция отображения	Операционная система коммутатора	Команда

QoS-стратегии	COS	`show qos acl edittbuffer`
		или
		`show qos acl info default-action {ip | ipx | mac | all}`
		`show qos acl info runtime {acl_name | all}`
		`show qos acl info config {acl_name | all} [edittbuffer [edit_buffer_index]]`

	IOS	`show class-map [class-name]`
		или
		`show policy-map policy-map-name`

Активность стратегии на интерфейсе	COS	`show qos acl map {config | runtime} {acl-name | mod/port | vlan | all}`
		или
		`show qos statistics {mod/port} | l3stats}`

	IOS	`show policy-map interface [type number | null interface-number | vlan vlan-id] [input | output]`

13.3: экспорт данных QoS

- Внутри коммутатора может осуществляться сбор данных QoS-статистики от отправителей, а также отправка этих данных устройству накопления.
- Возможность экспорта QoS-данных ограничена семейством коммутаторов Catalyst 6000.
- Статистические данные экспортируются через определенный UDP-порт или syslog-службу.
- Отправителем QoS-данных может быть один из перечисленных ниже объектов. При экспорте данных отображаемые поля, которые перечислены ниже, разделяются символом-разделителем.
  - **Порт коммутатора** — экспорт данных тип 1, сокет/порт, количество входящих пакетов, входящих байтов, исходящих пакетов, исходящих байтов и временная метка.
  - **Агрегированный регулировщик** — экспорт данных тип 3, имя регулировщика, количество профильных пакетов, количество непрофильных пакетов, превышающих CIR-скорость, количество непрофильных пакетов, превышающих PIR-скорость, а также временная метка.
  - **Таблица класса QoS-правил** — экспорт данных тип 4, имя таблицы класса, порт, VLAN-сеть или номер порта-канала, количество профильных пакетов, количество непрофильных пакетов, превышающих CIR-скорость, количество непрофильных пакетов, превышающих PIR-скорость, а также временная метка.

Конфигурирование функции

1. Определение способа отправки QoS-статистики.

а) Указание получателя для накопления статистических данных.

Система COS	`set qos statistics export destination` *host* `[`*port* `| syslog [`*facility-name severity*`]]`
Система IOS	`mls qos statistics-export destination` *{host-name	host-ip-address}* `{[port` *port number* `| syslog (facility` *facility-name*`) (severity` *severity*`)]}` (режим глобальной конфигурации)

Стандартно статистические данные получателю не отправляются. Статистика может быть отправлена получателю *host* (либо IP-адрес, либо имя узла) с использованием определенного UDP-порта (параметр *port*) или посредством службы syslog (UDP-порт 514). Если используется ключевое слово `syslog`, то syslog-средство (параметр *facility-name*) может быть задано с помощью слов `kern`, `user`, `mail`, `daemon`, `auth`, `lpr`, `news`, `uucp`, `cron`, `local0`, `local1`, `local2`, `local3`, `local4`, `local5`, `local6` (стандартное средство) и `local7`. Уровень важности syslog (параметр *severity*) — одно из следующих значений: `emerg`, `alert`, `crit`, `err`, `warning`, `notice`, `info` и `debug` (стандартная важность). Более подробная информация о службе syslog приведена в разделе "12.1: протоколирование событий".

б) Установка интервала экспорта данных *(необязательно)*.

Система COS	`set qos statistics export interval` *interval*
Система IOS	`mls qos statistics-export interval` *interval* (режим глобальной конфигурации)

QoS-статистика отправляется получателю через заданное параметром *interval* количество секунд (от 30 до 65 535 с; стандартно для операционной системы COS — 30 с, для системы IOS — 300 с).

Совет

При выборе интервала следует соблюдать осторожность. Если промежуток времени будет слишком большим, то счетчики QoS-статистики могут достичь своих максимальных значений, после чего будут установлены в ноль. Если будет выбрано очень малое время, значительно возрастет нагрузка на процессор коммутатора. Нужно начинать со стандартного значения и осуществлять настройку, учитывая влияние на процессор и счетчики.

в) Установка разделительного символа *(необязательно; только для IOS-коммутаторов)*

Система COS	Нет
Система IOS	`mls qos statistics-export delimiter` *character* (режим глобальной конфигурации)

Статистические данные QoS при необходимости могут быть разграничены с помощью определенного символа (параметр *character*, стандартно — символ <|>).

2. Включение функции сбора данных на коммутаторе.

Система COS	`set qos statistics export {enable | disable}`
Система IOS	`mls qos statistics-export` (режим глобальной конфигурации)

Стандартно QoS-статистика не накапливается и не экспортируется.

3. Сборка QoS-статистики от одного или нескольких отправителей

а) Выбор порта коммутатора (*необязательно*).

Система COS	`set qos statistics export port` *mod/port* `{enable | disable}`
Система IOS	`mls qos statistics-export` (режим конфигурирования интерфейса)

б) Выбор агрегированного регулировщика (*необязательно*).

Система COS	`set qos statistics export aggregate` *policer-name* `{enable | disable}`
Система IOS	`mls qos statistics-export aggregate-policer` *policer-name* (режим глобальной конфигурации)

Сбор статистики осуществляется от агрегированного регулировщика с именем, заданным параметром *policer-name* (текстовая строка). Регулировщик должен быть настроен согласно рекомендациям раздела "13.2: **Конфигурирование** средств QoS".

в) Выбор таблицы класса QoS (*необязательно, только для IOS-коммутаторов*).

Система COS	Нет
Система IOS	`mls qos statistics-export class-map` *classmap-name* (режим глобальной конфигурации)

При необходимости можно собирать статистические данные, касающиеся определенной части более сложной QoS-стратегии. Такая статистика собирается с помощью таблицы класса QoS с именем, заданным параметром *classmap-name* (текстовая строка). Таблица класса должна быть сконфигурирована согласно указаниям, приведенным в разделе "13.2: **Конфигурирование** средств QoS".

Пример конфигурирования экспорта данных QoS

Статистические данные QoS собираются в коммутаторе и отправляются накопительному узлу с адресом 192.168.111.14 при помощи syslog-средства `local6` с уровнем важности `debug` (стандартные установки). Данные накапливаются и отправляются каждые 300 секунд. Статистика собирается только для портов 3/1, 3/2 и агрегированного регулировщика с именем *MyPolicer*.

Система COS	`set qos statistics export destination 192.168.111.14 syslog`
	`set qos statistics export interval 300`
	`set qos statistics export enable`
	`set qos statistics export port 3/1-2 enable`
	`set qos statistics export aggregate MyPolicer enable`

Система IOS	`mls qos statistics-export destination 192.168.111.14 syslog`
	`mls qos statistics-export interval 300`
	`mls qos statistics-export`
	`interface gig 3/1`
	`mls qos statistics-export`
	`interface gig 3/2`
	`mls qos statistics-export`
	`mls qos statistics-export aggregate-policer MyPolicer`

Отображение информации об экспорте данных QoS

Для вывода на экран источников QoS-статистики используются приведенные ниже команды коммутатора.

Система COS	`show qos statistics export info`
Система IOS	`show mls qos statistics-export info`

13.4: управление QoS-функциями

- **Общая открытая служба правил (Common Open Policy Service — COPS)** — это протокол, который обеспечивает обмен данными и применение QoS-правил между устройствами в сети.

  COPS-управление поддерживается в коммутаторах семейства Catalyst 6000 с блоком коммутации третьего уровня (модуль PFC или PFC2)

- **Точка определения правил (Policy Decision Point — PDP)** поддерживает QoS-правила, которые будут соблюдаться внутри сети. Можно вести учет условий передачи трафика и на его основе менять реализацию правил. Типичным PDP-сервером может быть узел, использующий приложение Cisco *QoS Policy Manager* (QPM)

- **Точка применения правила (Policy Enforcement Point — PEP)** — устройство, содержащее физические интерфейсы, которыми управляет PDP-сервер. QoS-стратегии отправляются PEP, где они применяются в дискретном режиме. Обычно PEP-точками являются маршрутизаторы и коммутаторы.

  COPS-серверы конфигурируют QoS-стратегии для портов коммутатора PEP с помощью ролевых имен вместо физических номеров модулей и портов коммутатора. Это дает COPS-серверу возможность более абстрактного видения сетевого качества обслуживания. Всем портам коммутатора, которые будут контролироваться COPS-сервером, необходимо назначить произвольные ролевые имена.

- **Протокол резервирования ресурсов (Resource Reservation Protocol — RSVP)** используется для запросов и резервирования полосы пропускания для определенного потока данных. В сети устанавливается сквозное резервирование.

В методике RSVP-для-COPS используется COPS-механизм для сбора и управления RSVP-запросами.

- **Диспетчер полосы пропускания для выделенной подсети (Designated Subnet Bandwidth Manager — DSBM)** — это один коммутатор в сетевом сегменте, который выбирается для координации всей RSVP активности. Все устройства-клиенты RSVP-для-COPS пересылают RSVP-сообщения диспетчеру DSBM.

Конфигурирование функций QoS-администрирования

1. Выбор источника QoS-правил

   а) Установка отправителя для глобальной информации о стратегии

Система COS	`set qos policy-source {local | cops}`
Система IOS	Нет

Стандартно вся информация о QoS-правилах берется из локальной конфигурации коммутатора (`local`). Чтобы принять глобальную QoS-информацию от COPS-устройства, нужно использовать ключевое слово `cops`.

Совет

Глобальные QoS-правила включают в себя все таблицы преобразования DSCP, именованные и стандартные списки доступа, определения микропотоковых и агрегированных регулировщиков, расписание портов (привязка CoS-значений к очередям), предотвращение перегрузок (пороги очередей), весовые коэффициенты очередей, стандартное CoS-значение для порта и привязки списков доступа к портам коммутатора.

   б) Установка отправителя для определенных портов коммутатора.

Система COS	`set port qos policy-source` *mod/ports* `{local | cops}`
Система IOS	Нет

Стандартно вся информация о QoS-стратегии для каждого порта коммутатора взята из локальной конфигурации коммутатора (`local`). Чтобы получить от COPS-устройства конфигурацию QoS-стратегии для одного или нескольких портов коммутатора (*mod/ports*), нужно использовать ключевое слово `cops`.

2. Обмен данными с COPS-устройствами.

   а) Указание одного или нескольких PDP-серверов.

Система COS	`set cops server` *ipaddress* `[port] [primary] diff-serv`
Система IOS	Нет

PDP-сервер идентифицируется по IP-адресу (*ipaddress*). Можно задать необязательный номер TCP-порта (*port*). С помощью ключевого слова `primary` можно указать один основной сервер. Для указания резервного сервера команду нужно повторить один раз без ключевого слова `primary`.

б) Установка доменного имени COPS.

Система COS	`set cops domain-name` *domain*
Система IOS	Нет

Имя COPS-домена устанавливается с помощью параметра *domain* (текстовая строка длиной до 31 символа; стандартно — пустая строка). PDP-устройства способны контролировать PEP-устройства, находящиеся в том же домене.

в) Установка интервала повторного запроса COPS-сервера (необязательно).

Система COS	`set cops retry-interval` *initial incr max*
Система IOS	Нет

Для обмена данными с COPS-сервером используется интервал повторов. Коммутатор после первой неудавшейся попытки обмена ожидает в течение заданного времени (*initial*, от 0 до 65 535 секунд; стандартно — 30 с). После каждой последующей неудачной попытки длительность интервала повтора возрастает на величину *incr* (от 0 до 65 535 с; стандартно — 30 с) до максимального времени ожидания (*max*, от 0 до 65 535 секунд, стандартно — 30 с).

3. Назначение портам коммутатора COPS-ролей.

Система COS	`set port cops` *mod/port* `roles` *role1 [role2]...*
Система IOS	Нет

Стандартно все порты коммутатора не имеют ролей. Порту коммутатора можно назначить одно или несколько ролевых имен *role1*, *role2* (текстовая строка длиной до 31 символа). Общая длина всех ролевых имен, назначенных порту коммутатора, не может превышать 255 символов. В коммутаторе можно определить до 64 ролей.

4. Использование протокола RSVP с COPS-сервером (необязательно).

а) Включение поддержки протокола RSVP.

Система COS	`set qos rsvp {enable | disable}`
Система IOS	Нет

Стандартно поддержка протокола RSVP отключена.

б) Идентификация COPS PDP-сервера.

Система COS	`set cops server` *ipaddress [port]* `[primary] rsvp`
Система IOS	Нет

PDP-сервер идентифицируется по IP-адресу (*ipaddress*). Можно задать необязательный номер TCP-порта (*port*). С помощью ключевого слова `primary` можно указать один основной сервер. Для указания резервного сервера команду нужно повторить один раз без ключевого слова `primary`.

в) Участие коммутатора в управлении полосой пропускания.

Система COS	`set port rsvp mod/port dsbm-election {enable | disable} [dsbm-priority]`
Система IOS	Нет

Коммутатор может принимать участие в выборах диспетчера DSBM. Выборы разрешаются на указанных портах коммутатора (*mod/port*). Стандартно эта функция отключена. Приоритет коммутатора *dsbm-priority* (от 128 до 255, стандартно — 128) определяет вероятность победы в выборах, т. к. выигрывает устройство с наивысшим приоритетом.

г) Действие при отказе PDP-сервера (*необязательно*).

- Установка интервала ожидания DSBM (*необязательно*).

Система COS	`set qos rsvp policy-timeout timeout`
Система IOS	Нет

Если коммутатор избран в качестве диспетчера DSBM и теряет контакт с PDP-сервером, он продолжает выполнять функции диспетчера в течение заданного времени (*timeout*, от 1 до 65 535 минут; стандартно — 30 мин). В течение этого времени используются только кэшированные RSVP-стратегии. Если PDP-сервер не был восстановлен в течение времени ожидания, то коммутатор снова становится SBM-клиентом.

- Обработка RSVP-запросов в течение времени ожидания (*необязательно*).

Система COS	`set qos rsvp local-policy {forward | reject}`
Система IOS	Нет

Стандартно коммутатор продолжает пересылать (`forward`) все RSVP-сообщения о маршруте, даже если PDP-сервер недоступен. При необходимости можно использовать ключевое слово `reject` для отклонения всех новых или модифицированных RSVP-сообщений.

Отображение административной информации QoS

В табл. 13.10 перечислены команды коммутатора, которые можно использовать для отображения полезной информации об управлении QoS-функциями.

Таблица 13.10. Команды для отображения информации об управлении качеством обслуживания

Функция отображения	Операционная система коммутатора	Команда
Источник QoS-стратегии	COS	`show qos policy-source`
	IOS	Нет
Состояние COPS-сервера	COS	`show cops info [diff-serv]`
	IOS	Нет

Окончание табл. 13.10

Функция отображения	Операционная система коммутатора	Команда
COPS-роли порта коммутатора	COS	`show cops roles`
	IOS	Нет
RSVP-состояние	COS	`show qos rsvp info`
	IOS	Нет
Активные RSVP-потоки	COS	`show qos rsvp flow-info`
	IOS	Нет

Дополнительная литература

Рекомендуемые источники предоставляют дополнительную информацию по темам, рассматриваемым в этой главе.

Шринивас Весешва. *Качество обслуживания в сетях IP*. М.: "Вильямс", 2003.

Рабочая группа IETF по разработке дифференцированных служб (Differentiated Services (DiffServ) IETF Working Group): `www.ietf.org/html.charters/diffserv-charter.html`

Протокол COPS (The COPS Protocol). RFC 2748 `www.faqs.org/rfcs/rfc2748.html`

Протокол RSVP (The RSVP Protocol): `www.isi.edu/div7/rsvp/rsvp.html`.

QoS-стратегии в коммутаторе Catalyst 6000 (QoS Policing on the Catalyst 6000): `www.cisco.com/warp/customer/473/102.html`.

Реализация QoS в коммутаторах семейства Catalyst 6000: исходящее расписание в коммутаторе Catalyst 6000 с модулем PFC в гибридном режиме (QoS on Catalyst 6000 Family Switches. Output Scheduling on the Catalyst 6000 with PFC Using Hybrid Mode). `www.cisco.com/warp/public/473/60.html`.

Реализация QoS в коммутаторах семейства Catalyst 6000: исходящее расписание в коммутаторе Catalyst 6000 с модулем PFC в собственном режиме IOS (QoS on Catalyst 6000 Family Switches: Output Scheduling on the Catalyst 6000 with PFC Using IOS Native Mode): `www.cisco.com/warp/public/473/78.html`.

Основные понятия RACL/VACL/QoS аппаратных ресурсов ACL в коммутаторах семейства Catalyst 6000 (Understanding RACL/VACL/QoS ACL Hardware Resources in Catalyst 6000 Family Switches): `www.cisco.com/warp/public/473/79.html`.

В этой главе...

- **14.1: голосовые порты.** В этом разделе описываются команды, необходимые для конфигурирования коммутируемых Ethernet-портов с целью обеспечения поддержки IP-телефонии.
- **14.2: QoS-параметры при передаче голосовых данных.** В этом разделе представлены основные принципы и рекомендации по конфигурированию, направленные на обеспечение сквозной поддержки уровня QoS в территориальной сети.
- **14.3: голосовые модули.** В разделе описываются этапы конфигурирования модулей голосового шлюза в коммутаторах Catalyst серий 4000 и 6000.

Глава 14

Поддержка передачи голосовых данных

14.1: голосовые порты

- Линейным питанием обеспечиваются следующие активные устройства:
  - устройство с переменной активностью может быть обнаружено, как только активизируется порт коммутатора;
  - активное устройство образует замкнутые цепи на прием и передачу таким образом, что коммутатор обнаруживает собственный тестовый сигнал частотой 340 кГц;
  - если голосовое устройство присутствует, то на порт подается питание; если на порту обнаружено обычное Ethernet-устройство, питание не подается;
  - линейное питание (inline power) — постоянный ток с напряжением в 46 В — подается через пары 2 и 3 (контакты 1, 2 и 3, 6 разъема RJ-45).
- Линейное питание доступно в следующих модулях коммутатора:
  - 48-портовый модуль линейного питания 10/100 Ethernet коммутатора Catalyst 6000 (Catalyst 6000 Inline Power 10/100 Ethernet Switch Module — код WS-X6348-RJ45V);
  - 48-портовый модуль линейного питания 10/100 Ethernet коммутатора Catalyst 4000 (Catalyst 4000 Inline Power 10/100 Ethernet Switch Module — код WS-X4148-RJ45V) с дополнительной платой (auxiliary DC power shelf) постоянного тока и модулем подачи питания (power entry module);
  - Catalyst 3524XL.
- В описанных ниже ситуациях питание также может быть подано через вышеупомянутую 48-портовую монтажную панель питания (power patch panel — код WS-PWR-PANEL)
  - Устройство с переменной активностью не обнаружено.
  - Активные устройства можно подключить к настенному адаптеру питания и к монтажной панели питания. Монтажная панель используется устройствами в качестве резервного источника питания.

- Постоянный источник питания с напряжением 48 В подключается через пары 1 и 4 (контакты 4, 5 и 7, 8 разъема RJ-45)

- Коммутатор Catalyst способен передавать IP-телефону Cisco инструкции о том, каким образом представлять фреймы с его голосовых портов и портов данных. Эта операция осуществляется посредством сообщений *протокола обнаружения устройств Cisco (Cisco Discovery Protocol — CDP)*
- Коммутатор и телефон могут обмениваться данными через магистральный канал 802.1Q. Голосовой трафик передается в сети с отдельным идентификатором VVID (voice VLAN ID — идентификатор голосовой VLAN-сети). CoS-информация *(Class of Service — класс обслуживания)* для передачи голосовых данных может распространяться через этот магистральный канал.
- IP-телефон Cisco в процессе инициализации проходит описанные ниже этапы.

1. При необходимости линейное питание обнаруживается коммутатором.
2. Телефон запускает CDP-обмен. Коммутатору отправляется фактическая величина необходимой мощности, а телефону отправляется VVID-номер. Телефон также может принимать инструкции о том, как расширить доверительные границы качества обслуживания (QoS-trust boundary).
3. Если необходимо обеспечить поддержку VVID, между телефоном и коммутатором согласовывается специальный 802.1Q-канал. В коммутаторах Catalyst серий 4000 и 6000 магистральный канал согласовывается посредством сообщений *динамического магистрального протокола (Dynamic Trunk Protocol — DTP)*. В коммутаторах Catalyst 3500XL протокол DTP не поддерживается, поэтому магистральный канал необходимо конфигурировать вручную.
4. Выполняется DHCP-запрос.
5. Телефону отправляется DHCP-ответ, содержащий IP-адрес и адрес TFTP-сервера (DHCP-параметр 150).
6. Обеспечивается соединение с TFTP-сервером в целях получения конфигурационного файла для телефона. Также предоставляется список серверов управления вызовами (CallManager) Cisco.
7. Осуществляется регистрация в CallManager-сервере. Предоставляется *номер каталога (Directory Number — DN)*, с тем чтобы можно было инициировать и принимать вызовы.

Конфигурирование функции

1. Использование линейного питания для IP-телефонов Cisco.

   а) Установка стандартного распределения питания (*необязательно*).

Система COS	`set inlinepower defaultallocation value`
Система IOS	Нет

Мощность, выделяемая порту коммутатора, согласовывается с активным устройством. Стандартно на каждый порт коммутатора подается 10,0 Вт (постоянный ток 0,24 А напряжением 42 В). Стандартное значение можно изменить с помощью параметра *value* (2000–12500 мВт, стандартно — 10000 мВт).

Совет

Стандартное распределение мощности основано на общей мощности, предоставляемой источниками питания коммутатора. При изменении стандартного распределения необходимо убедиться, что активизация всех активных устройств не приведет к превышению доступной мощности. Общую доступную мощность можно определить с помощью COS-команды `show environment power`. Определить, какие порты коммутатора превышают доступную мощность, можно с помощью COS-команды `show port inlinepower`.

Хотя коммутатор первоначально обеспечивает устройству стандартное распределение мощности, ее величина может быть изменена по согласованию при обмене CDP-сообщениями.

б) Обнаружение устройства с линейным питанием (*необязательно*).

Система COS	`set port inlinepower` *mod/port* `{off | auto}`
Система IOS	`power inline {auto | never}` (режим конфигурирования интерфейса)

Стандартно коммутатор пытается обнаружить на порту устройство с линейным питанием (режим `auto`). Использование ключевых слов `off` (COS) и `never` (IOS) приводит к отключению функции.

Внимание!

После обнаружения активного устройства и подачи питания в порт коммутатор в течение четырех секунд ожидает инициализации устройства и установки канала. Если этого не произошло, питание с данного порта коммутатора снимается.

Если в течение четырехсекундной задержки отключить от порта активное устройство и подключить вместо него обычное Ethernet-устройство, то питание будет подаваться и далее, что может привести к повреждению устройства. Перед сменой устройств на портах коммутатора рекомендуется сделать паузу по крайней мере на 10 секунд.

2. Конфигурирование VLAN-сетей с IP-телефонов.

Совет

IP-телефон Cisco может использовать магистральный канал 802.1Q для транспортировки пакетов от двух VLAN-сетей: голосовой VLAN-сети (голосовые пакеты) и собственной сети VLAN (немаркированные пакеты данных). Стандартно IP-телефоны Cisco транспортируют как голосовые, так и пакеты данных от подключенных устройств через собственную VLAN-сеть. Данные при этом не маркируются.

После того как коммутатор был настроен на отправку инструкций IP-телефону для поддержки какого-либо VVID-номера, коммутатор и телефон должны использовать между собой магистральный канал 802.1Q. В коммутаторах Catalyst 3500XL согласование магистрального канала не поддерживается, и канал 802.1Q необходимо сконфигурировать вручную. IP-телефон на своем конце соединения использует этот тип магистрального канала автоматически.

Коммутаторы Catalyst 4000 и 6000 при помощи протоколов CDP и DTP согласовывают с IP-телефоном специальный канал 802.1Q. Как только телефон будет обнаружен, порт коммутатора станет портом VLAN-доступа сети VLAN 2 (*vlan2-access port*), поддерживающим только две VLAN-сети — передачи голоса и данных. Порт не будет показан как порт в магистральном режиме при выводе информации с помощью команды `show`

`trunk`. Фактически не имеет значения, какой режим магистрального соединения (*auto*, *desirable*, *on* или *off*) сконфигурирован на порту. Специальный магистральный канал будет согласовываться с помощью протокола DTP. Следует убедиться, что при конфигурировании магистрального канала не было использовано ключевое слово `nonegotiate`, поскольку в таком случае не будут отправляться или приниматься DTP-сообщения, и автоматическая установка магистрального канала не произойдет.

Также в магистральных каналах IP-телефонов автоматически обеспечивается поддержка протокола *распределенного связующего дерева (Spanning Tree Protocol — STP)*. С помощью команды `show spanntree` можно отобразить STP-состояние обеих VLAN-сетей на магистральном канале.

а) Использование VLAN-сети для передачи данных (*необязательно*).

- Идентификация VLAN-сети доступа порта коммутатора (*необязательно*)

Система COS	`set vlan` *vlan-id mod/ports*
Система IOS	`switchport access vlan` *vlan-id* (режим конфигурирования интерфейса)

Можно настроить порты коммутатора на поддержку персональных компьютеров и IP-телефонов. Если к порту коммутатора подключается обычный узел (не IP-телефон), VLAN-сети доступа следует задать идентификатор *vlan-id* (от 1 до 1000 или от 1025 до 4094). Если к порту подключен персональный компьютер, поддерживается только VLAN-сеть доступа, а специальный магистральный канал не согласовывается. Более подробная информация приведена в разделе "6.1: конфигурация VLAN-сети".

- Определение собственной VLAN-сети порта коммутатора

Система COS	`set vlan` *vlan-id mod/ports*
Система IOS	`switchport trunk native vlan` *vlan-id* (режим конфигурирования интерфейса)

Данные из порта доступа на IP-телефоне транспортируются по собственной VLAN-сети (немаркированные данные) специального магистрального 802.1Q-канала. Следовательно, необходимо указать номер собственной VLAN-сети как *vlan-id* (от 1 до 1000 или от 1025 до 4094). В COS-коммутаторах для этой цели применяется такая же команда, как и для настройки VLAN-сети доступа порта коммутатора.

2. Передача телефону инструкций на транспортировку данных и голоса (*необязательно*).

а) Использование магистрального 802.1Q-канала с голосовой VLAN-сетью (*необязательно*)

Система COS	`set port auxiliaryvlan` *mod[/port] vlan-id*
Система IOS	`switchport voice vlan` *vlan-id* (режим конфигурирования интерфейса)

С помощью этой команды IP-телефону передаются инструкции на использование магистрального 802.1Q-канала. Голосовые фреймы маркируются VLAN-идентификатором (*vlan-id*, от 1 до 4094 — в COS или от 1 до 1001 —

в IOS), тогда как фреймы из порта данных телефона отправляются немаркированными (по собственной VLAN-сети). CoS-значения голосовых фреймов транспортируются в поле приоритета 802.1p.

б) Использование магистрального 802.1Q-канала без голосовой VLAN-сети (*необязательно*).

Система COS	`set port auxiliaryvlan` *mod[/port]* `dot1p`
Система IOS	`switchport voice vlan dot1p` (режим конфигурирования интерфейса)

С помощью этой команды IP-телефону передаются инструкции на использование магистрального 802.1Q-канала и поля CoS-приоритета 802.1p, однако все голосовые данные помещаются в несуществующую сеть (VLAN 0). Фреймы из портов данных телефона помечаются немаркированными (по собственной VLAN-сети). Это позволяет распространять сведения о приоритете голосовых данных без использования отдельной голосовой VLAN-сети.

в) Использование магистрального 802.1Q-канала без VLAN-информации (*необязательно*).

Система COS	`set port auxiliaryvlan` *mod[/port]* `untagged`
Система IOS	`switchport voice vlan untagged` (режим конфигурирования интерфейса)

IP-телефон получает инструкции отправлять немаркированными все голосовые фреймы по собственной VLAN-сети, в результате чего инкапсуляция 802.1Q не используется, и распространение какой-либо CoS-информации о приоритете 802.1p невозможно.

г) Отключение передачи инструкций телефону (*необязательно*).

Система COS	`set port auxiliaryvlan` *mod[/port]* `none`
Система IOS	`switchport voice vlan none` (режим конфигурирования интерфейса)

Коммутатор не будет предоставлять IP-телефону идентификатор VVID. Такая конфигурация является стандартной. Телефон не получит сведений о голосовой VLAN-сети, и оба типа фреймов — как голосовые, так и фреймы данных — будут отправляться порту коммутатора через одну и ту же VLAN-сеть доступа.

д) Конфигурирование магистрального 802.1Q-канала вручную (*только для коммутаторов Catalyst 3500XL*).

- Выбор 802.1Q-инкапсуляции.

Система COS	Нет
Система IOS	`switchport trunk encapsulation dot1q` (режим конфигурирования интерфейса)

С помощью этой команды на интерфейсе коммутатора активизируется 802.1Q-инкапсуляция. IP-телефон автоматически использует и ожидает установки магистрального 802.1Q-канала.

- Размещение в магистральном канале голосовой VLAN-сети и VLAN-сети для передачи данных (*необязательно*)

Система COS	Нет
Система IOS	`switchport trunk allowed vlan` *vvid*`,`*pvid* (режим конфигурирования интерфейса)

Стандартно в магистральном канале разрешены все сконфигурированные VLAN-сети. Чтобы предотвратить использование каким-либо широковещательным трафиком в любых VLAN-сетях излишней полосы пропускания, можно ограничить разрешение VLAN-сети только голосовой VLAN (с номером *vvid*) и VLAN-сетью передачи данных (номер *pvid*).

- Включение магистрального канала.

Система COS	Нет
Система IOS	`switchport mode trunk` (режим конфигурирования интерфейса)

3. Оптимизация порта коммутатора для IP-телефона (*необязательно*).

Совет

COS-коммутатор способен осуществлять действия, описанные в последующих конфигурационных этапах, с помощью одной команды `set port host` *mod/port*. Эта команда фактически отключает магистральный режим на порту коммутатора. Однако независимо от этого порт коммутатора и IP-телефон продолжают использовать специальную форму 802.1Q-транкинга.

а) Отключение поддержки EtherChannel.

Система COS	`set port channel` *mod/port* `mode off`
Система IOS	`no channel-group` (режим конфигурирования интерфейса)

Поддержка динамической EtherChannel-конфигурации с помощью *протокола объединения портов (Port Aggregation Protocol — PAgP)* отключена, что позволяет сэкономить около 10 секунд при активизации порта. Дополнительные сведения представлены в разделе "9.4: порты EtherChannel".

б) Включение STP-функции PortFast

Система COS	`set spantree portfast` *mod/port* `enable`
Система IOS	`spanning-tree portfast` (режим конфигурирования интерфейса)

Порт коммутатора настраивается на ускоренный запуск STP, минуя STP-состояния *прослушивания (listening)* и *самообучения (learning)*. Этот порт может быть немедленно переведен в состояние *передачи (forwarding)*. Более подробная информация приведена в разделе "7.3: точная настройка конвергенции распределенного связующего дерева".

Пример конфигурирования функции

В нашем примере производится конфигурирование коммутатора Catalyst для поддержки на его порту IP-телефона. Коммутатор поддерживает линейное питание, но порт коммутатора может подключаться к обычному персональному компьютеру или IP-телефону Cisco.

Порт настраивается на автоматическое обнаружение устройства, поддерживающего линейное питание. VLAN-идентификатор доступа или порта (PVID) устанавливается равным VLAN-номеру 55. Если к порту коммутатора непосредственно подключается персональный компьютер, то все фреймы данных транспортируются через VLAN-сеть доступа. Если к порту подключен IP-телефон, то согласовывается магистральный 802.1Q-канал для двух VLAN-сетей. Фреймы данных персонального компьютера, подключенного к телефону, транспортируются немаркированными через собственную сеть VLAN 55 по магистральному каналу. Голосовые фреймы, направленные телефону и от него, маркируются и транспортируются через голосовую или дополнительную VLAN-сеть (VVID) 200 по магистральному каналу.

Порт коммутатора также конфигурируется для сокращения задержек инициализации, вызванных протоколами PAgP и STP. Эти настройки необязательны, но позволяют IP-телефону избежать ожидания перед загрузкой конфигурационных данных телефона.

Система COS	`set port inlinepower 4/1 auto`
	`set vlan 55 4/1`
	`set port auxiliaryvlan 4/1 200`
	`set port host 4/1`

Система IOS	`interface fastethernet 0/1`
	`power inline auto`
	`switchport access vlan 55`
	`switchport trunk native vlan 55`
	`switchport voice vlan 200`
	`switchport trunk encapsulation dot1q`
	`switchport mode trunk`
	`no channel-group`
	`spanning-tree portfast`

Отображение информации о голосовых портах

В табл. 14.1 приведены некоторые команды коммутаторов, с помощью которых можно получить полезные сведения о голосовых портах.

Таблица 14.1. Команды коммутатора для отображения информации о голосовых портах

Функция отображения	Операционная система коммутатора	Команда
Состояние линейного питания	COS	`show port inlinepower [mod[/port]]`

Функция отображения	Операционная система коммутатора	Команды

	IOS	`show power inline [interface-id] [actual | configured]`
		или
		`show cdp neighbor [interface-id] detail`

VLAN-сеть доступа, собственная и голосовая	COS	`show port mod/port`
		или
		`show trunk mod/port detail`

	IOS	`show interface [interface-id] switchport`
Обнаруженные устройства	COS	`show cdp neighbor mod/port [detail]`
	IOS	`show cdp neighbor [interface-id] [detail]`

14.2: QoS-параметры при передаче голосовых данных

Чтобы обеспечить соответствующую доставку голосового трафика в иерархической коммутируемой сети, необходимо следовать нескольким практическим рекомендациям, связанным с параметрами качества обслуживания (QoS). Рассмотрим диаграмму сети на рис. 14.1.

- Уровень доступа.
  - Доверительную границу QoS следует устанавливать как можно ближе к конечным устройствам (на уровне доступа).
  - Рекомендуется разрешить IP-телефону поддерживать доверительную границу для подключенных PC-станций. IP-телефон должен быть устройством, которому доверяют.
  - Персональным компьютером, использующим приложение Cisco SoftPhone, не должны доверять. Вместо этого рекомендуется классифицировать входящий трафик и устанавливать значения CoS и *кодовой точки дифференцированных служб (Differentiated Services Code Point — DSCP)*
  - Обычным PC-станциям без возможности передачи голосовых данных не должны доверять (значения CoS и ToS *(Type of Service — тип обслуживания)* устанавливаются равными нулю).
  - В коммутаторах Catalyst 6000 доверие к порту может зависеть от VLAN сети и применяться к голосовой VLAN-сети на всех портах, которым доверяют.
  - Следует модифицировать таблицы преобразования CoS- и ToS-значений в DSCP так, чтобы значение 3 преобразовывалось в DSCP 26 (AF31), а значение 5 — в DSCP 46 (EF) тех, где это возможно.
  - Если возможно, внешние каналы к уровню распределения и основному уровню должны доверять DSCP-значениям.

- Механизм планирования для порта должен быть таким, чтобы исходящие голосовые фреймы с CoS-значением 5 назначались очереди с более высоким приоритетом. Фреймы с CoS 5 автоматически назначаются исходящей очереди со строгим приоритетом.

- Уровни распределения и основной.
  - Если DSCP-значения могут контролироваться коммутаторами уровня доступа, то им нужно доверять.
  - Если коммутаторы уровня доступа являются устройствами исключительно второго уровня и неспособны классифицировать или маркировать фреймы по основании DSCP, DSCP-значения для голосовых фреймов необходимо устанавливать в коммутаторах вышестоящего уровня. Это можно сделать в голосовой VLAN-сети для портов, которым доверяют и которые сконфигурированы на основе VLAN.
  - Следует модифицировать таблицы преобразования CoS- и ToS-значений в DSCP так, чтобы значение 3 соответствовало коду DSCP 26 (AF31), а значение 5 — коду DSCP 46 (EF) везде, где это возможно.
  - Планирование должно быть таким, чтобы исходящие голосовые фреймы с CoS-значением 3 назначались очереди с более высоким приоритетом. Фреймы со значением CoS, равным 5, автоматически назначаются исходящей очереди со строгим приоритетом.

Внутри сети можно использовать несколько голосовых протоколов.

- **Протоколы управления передачей голоса** используются для регистрации и установки вызовов:
  - *упрощенный протокол управления клиентом (Skinny Client Control Protocol — SCCP)*, который также называется *простым протоколом управления клиентом (Simple Client Control Protocol)*;
  - протокол H.323;
  - *протокол создания сеанса (Session Initiation Protocol — SIP)*;
  - *протокол управления шлюзом (Media Gateway Control Protocol — MGCP)*;
  - протокол Megaco, или H.248.
- **Транспортный протокол реального времени (Real-Time Transport Protocol — RTP)** представляет собой UDP-инкапсуляцию пакетов действительного носителя голоса. Во всех протоколах передачи голосовых данных после установки вызова в качестве транспортного механизма используется протокол RTP.

Указанные протоколы передачи голосовых данных используют номера UDP- или TCP-портов, приведенные в табл. 14.2. Эти значения могут оказаться полезными, когда необходимо классифицировать голосовой трафик для служб QoS в коммутаторе Catalyst. Все протоколы управления голосовыми вызовами следует маркировать кодом CoS 3 или DSCP 26 (AF31). RTP-пакеты носителя голоса *везде* следует маркировать значением CoS 5 или DSCP 46 (EF), чтобы обеспечить их своевременную доставку. Маркирование RTP-пакета обычно осуществляется отправителем.

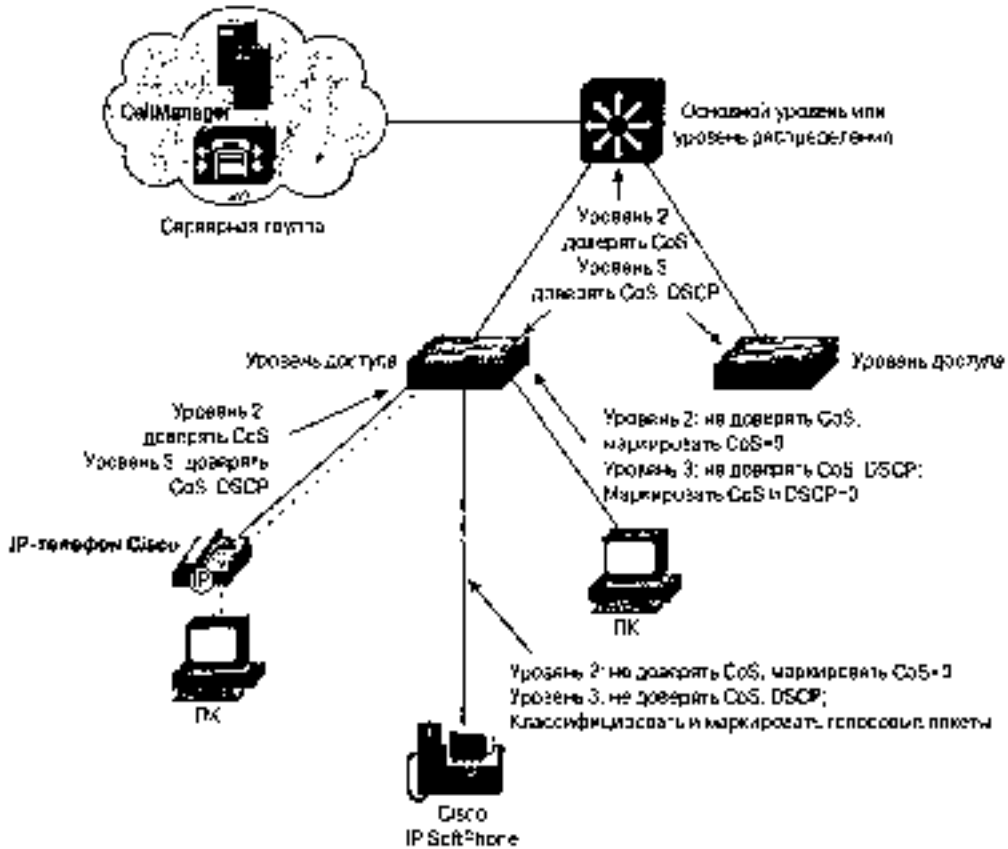


Рис. 14.1. Понятия QoS-доверия в коммутируемой сети

Таблица 14.2. Номера портов протоколов передачи голосовых данных

Голосовой протокол	Порт	Описание
Простой	TCP 2000	Skinny Client Control Protocol (SCCP)
	TCP 2001	Простой протокол управления станцией (Skinny Station Protocol — SSP)
	TCP 2002	Простой протокол шлюза (Skinny Gateway Protocol — SGP)
H.323	TCP 1718	Сообщения Gatekeeper
	TCP 1719	Сообщения Gatekeeper RAS
	TCP 1720	Управление вызовом H.225
	TCP с 11000 по 11999	Протокол H.245
SIP	UDP/TCP 5060	Стандартные порты сервера, также могут выбираться произвольно
MGCP	TCP 2427	Агенты вызова к шлюзу
	TCP 2727	Шлюз к агентам вызова

Голосовой протокол	Порт	Описание
Megaco	UDP/TCP 2944	Текстовые сообщения управления вызовом
H.248	UDP/TCP 2945	Двоичные сообщения управления вызовом
RTP	UDP-порт согласуется сигнальным протоколом голосового вызова	Транспортировка полезной голосовой нагрузки

Конфигурирование уровня доступа

Совет

Представленные ниже команды разделены согласно используемым платформам коммутации. Коммутаторы третьего уровня имеют блок коммутации третьего уровня, поддерживаемый Catalyst 6000 с платой *PFC* (*Policy Feature Card* — функциональная плата политик) или *PFC2*. Коммутаторы второго уровня имеют блок коммутации второго уровня, поддерживаемый на платформах Catalyst 4000 и 5000. Модели коммутаторов Catalyst 2900XL и 3500XL обозначены как 3500.

1. Установка доверительной границы на уровне доступа (*необязательно*).

   а) Доверие QoS от IP-телефона Cisco (*необязательно*)

COS третьего уровня	`set port qos` *mod/port* `vlan-based` `set port qos` *mod/port* `trust trust-cos`

IOS третьего уровня	`mls qos vlan-based`
	`mls qos trust cos`
	(обе команды вводятся в режиме конфигурирования интерфейса)

COS второго уровня	`set port qos` *mod/port* `trust trust-cos`
IOS второго уровня	`mls qos trust cos` (режим конфигурирования интерфейса)

IOS 3500	`switchport priority default 0`
	`no switchport priority override`
	`switchport priority extend cos 0`
	(все команды вводятся в режиме конфигурирования интерфейса)

К голосовому трафику от всех IP-телефонов в общей голосовой VLAN-сети может быть применено единое QoS-правило. Это возможно только в коммутаторах третьего уровня. В противном случае входящим CoS-значениям можно доверять, когда IP-телефоны классифицируют и маркируют классы собственных голосовых портов и портов доступа к данным. IP-телефоны получают инструкции по управлению QoS на этапе 3.

Совет

IP-телефон Cisco маркирует свои SCCP-пакеты управления голосом значениями CoS 3, ToS 3 и DSCP 26 (AF31). RTP-пакеты носителя голоса маркируются метками CoS 5 ToS 5

и DSCP 46 (EF). Эти пакеты транспортируются во фреймах в голосовой VLAN-сети (VVID) магистрального 802.1Q-канала.

Если имеются соответствующие инструкции, то IP-телефон также маркирует трафик от своего порта коммутатора доступа. Стандартно эти фреймы транспортируются немаркированными через собственную VLAN-сеть магистрального 802.1Q-канала и имеют ToS- и DSCP-значения, равные нулю.

б) Не доверять QoS-значениям от PC-станции, использующей Cisco SoftPhone (*необязательно*).

COS третьего уровня	`set port qos mod/port cos` `set port qos mod/port trust untrusted`

IOS третьего уровня	`mls qos cos 0`
	`no mls qos trust`
	(режим конфигурирования интерфейса)

COS второго уровня	`set port qos mod/port cos 0` `set port qos mod/port trust untrusted`

IOS второго уровня	`mls qos cos 0`
	`no mls qos trust`
	(режим конфигурирования интерфейса)

IOS 3500	`switchport priority default 0`
	`switchport priority override`
	(обе команды вводятся в режиме конфигурирования интерфейса)

Несмотря на то что персональные компьютеры, на которых используется приложение SoftPhone, создают пакеты управления голосом и пакеты данных носителя, другие работающие приложения могут попытаться изменить CoS-значения в обычных (т. е. которые не несут голосовых данных) пакетах. Поэтому не следует доверять QoS-информации, поступающей от PC-станций. Рекомендуется установить порты коммутатора в состояние недоверия и настроить коммутаторы третьего уровня в имеющемся QoS-домене на классификацию и соответствующее маркирование пакетов управления голосом (voice control) и пакетов носителя (bearer packet).

Совет

Приложение Cisco SoftPhone маркирует свои пакеты управления голосом SCCP-значениями CoS 0,ToS 0 и DSCP 0 (стандартно). RTP-пакеты носителя голоса маркируются значениями CoS 5,ToS 5 и DSCP 46 (EF). Эти пакеты транспортируются через VLAN-сеть доступа немаркированными, ввиду того что собственный магистральный канал не используется.

в) Не доверять QoS-параметрам от узлов, передающих только обычные данные (*необязательно*).

COS третьего уровня	`set port qos mod/port cos 0` `set port qos mod/port trust untrusted`

IOS третьего уровня	`mls qos cos 0`
	`no mls qos trust`
	(режим конфигурирования интерфейса)

COS второго уровня	`set port qos mod/port cos 0` `set port qos mod/port trust untrusted`

IOS второго уровня	`mls qos cos 0`
	`no mls qos trust`
	(режим конфигурирования интерфейса)

IOS 3500	`switchport priority default 0`
	`switchport priority override`
	(режим конфигурирования интерфейса)

Немаркированные фреймы или фреймы, которые не соответствуют ни одному *списку управления доступом (Access Control Lists — ACL)* с QoS-классификацией, будут маркироваться CoS-значением 0. Это также приводит к тому, что входящие DSCP-значения преобразовываются в 0 с помощью таблицы преобразования CoS-DSCP.

2. Регулировка входного CoS-DSCP-преобразования (*необязательно; только для коммутаторов третьего уровня*).

Система COS	`set qos cos-dscp-map 0 8 16 26 32 46 48 56` `set qos ipprec-dscp-map 0 8 16 26 32 46 48 56`

Система IOS	`mls qos map cos-dscp 0 8 16 26 32 46 48 56`
	`mls qos map ip-prec-dscp 0 8 16 26 32 46 48 56`
	(режим глобальной конфигурации)

Можно внести такие настройки преобразования так, чтобы значение CoS 3 преобразовывалось в код DSCP 26 (AF31), а значение CoS 5 — в код DSCP 46 (EF). Значения, принятые по умолчанию, несколько отличаются и не являются стандартными значениями, ожидаемыми для голосового трафика.

3. Расширение QoS-доверия в IP-телефоне (*необязательно*).

а) Установка доверия для порта доступа телефона.

Система COS	`set port qos mod/ports trust-ext {trusted | untrusted}`
Система IOS	`switchport priority extend {trust | none}` (режим конфигурирования интерфейса)

IP-телефон Cisco имеет встроенный порт, к которому можно подключить персональный компьютер. Этот порт стандартно функционирует в режиме `untrusted` (или `none` в IOS), что приводит к установке нулевых значений CoS и IP-приоритета для входящих фреймов. Чтобы разрешить персональной станции маркировать свои пакеты значениями IP-приоритета, следует установить режим `trusted` (`trust` в системе IOS).

б) Установка стандартного CoS значения порта доступа телефона.

Система COS	`set port qos mod/ports cos-ext cos-value`
Система IOS	`switchport priority extend cos cos-value` (режим конфигурирования интерфейса)

Если порт доступа телефона настроен для работы в режиме `untrusted`, то телефон устанавливает для всех входящих фреймов данных CoS-значение, равное параметру `cos-value` (от 0 до 7; стандартное значение равно нулю).

4. Доверять DSCP-информации на внешних портах (*только для коммутаторов третьего уровня*).

Система COS	`set port qos` *mod/ports* `trust trust-dscp`
Система IOS	`mls qos trust dscp` (режим конфигурирования интерфейса)

Поскольку коммутаторы основного уровня и уровня распределения также находятся внутри QoS-домена и соответствующим образом настроены для удовлетворения QoS-требований, можно с уверенностью предположить, что любые сведения о качестве обслуживания, поступающие от этих устройств, проверены и отретушированы для удовлетворения правил QoS. Поэтому такой информации можно доверять на внешних портах коммутатора уровня доступа.

5. Применение QoS-правил к (входящему) трафику (*необязательно; только для коммутаторов третьего уровня*).

а) Указание трафика с помощью ACL.

Система COS	`set qos acl ip` *acl-name* `dscp 26 tcp any any range 2000 2002` `set qos acl ip` *acl-name* `trust-cos ip any any`

Система IOS	`ip access-list extended` *acl-name*
	(режим глобальной конфигурации)
	`permit tcp any any range 2000 2002 dscp 26`
	`exit`

В данном случае TCP-порты 2000, 2001 и 2002 протокола управления голосом SCCP соответствуют списку доступа. Таким фреймам задается DSCP-значение 26 (AF31), даже если оно уже установлено. ACL-список проверки также является необходимым для того, чтобы можно было установить CoS-доверие на портах коммутатора, сконфигурированных с помощью команды `set port qos trust trust-cos`.

Если используются другие протоколы управления голосом, то можно изменить ACL-список для соответствия необходимым номерам портов.

б) Определение QoS-стратегии (*только для системы IOS третьего уровня*).

Система COS	Нет
Система IOS	`policy-map` *policy-name* (режим глобальной конфигурации) `class` *class-name* `access-group` *acl-name* `trust cos`

В этой стратегии используется класс для проверки трафика из ACL списка. Для соответствующего трафика COS-значения впоследствии являются доверенными.

в) Применение QoS-правила к голосовой VLAN-сети.

Система COS	`commit qos acl acl-name` `set qos acl map acl-name voice-vlan`

Система IOS	`interface vlan voice-vlan`
	(режим глобальной конфигурации)
	`service-policy input policy-name`
	(режим конфигурирования интерфейса)

Можно применить QoS-правило ко всем портам, поддерживающим голосовую VLAN сеть. Такой подход является эффективным способом использования QoS-стратегии в одной определенной VLAN-сети внутри магистрального канала

6. Конфигурирование планировщика для голосовых потоков на выходных портах.

Коммутаторы Catalyst 2900XL и 3500XL на своих выходных портах используют фиксированное планирование. Фреймы управления голосовыми данными с CoS-значением 5 направляются в низкоприоритетную очередь (очередь 1), тогда как фреймы с CoS-значением 5 отправляются в высокоприоритетную очередь (очередь 2). Очереди со строгим приоритетом не существует

COS третьего уровня	`set port qos mod/port port-based`
	`set qos map 1p2q2t tx 2 1 cos 5`
	`set qos map 2q2t tx 2 1 cos 5`

IOS третьего уровня	`no mls qos vlan-based`
	`wrr-queue cos-map 2 1 5`
	(режим конфигурирования интерфейса)

COS второго уровня	`set qos map 1p2q2t tx 2 1 cos 5` `set qos map 2q2t tx 2 1 cos 5`
IOS второго уровня	`wrr-queue cos-map 2 1 5` (режим конфигурирования интерфейса)
IOS 3500	Нет

Стандартно все фреймы с CoS-значением, равным пяти, отправляются в очередь со строгим приоритетом. Фреймы с CoS-значением, равным трем, отправляются в низкоприоритетную очередь. Таблица планировщика позволяет обеспечить отправку фреймов управления голосом (CoS 5) в высокоприоритетную очередь и их первоочередную по отношению к остальному трафику обработку.

Конфигурирование устройств уровня распределения и основного уровня

1. Установка доверительной границы.

а) Доверять QoS-информации на основе VLAN из коммутатора второго уровня на уровне доступа (*необязательно; только для коммутаторов третьего уровня*).

Коммутатор второго уровня, расположенный на уровне доступа, может классифицировать и маркировать трафик только на основании CoS-значений это-

рого уровня. Кроме того, QoS-требования применяются к голосовой VLAN-сети, в которой транспортируется трафик IP-телефонии. Коммутатор уровня распределения или основного уровня впоследствии может применить правила QoS непосредственно к голосовой VLAN-сети.

COS третьего уровня	`set port qos mod/port vlan-based` `set port qos mod/port trust trust-cos`

IOS третьего уровня	`mls qos vlan-based`
	`mls qos trust cos`
	(режим конфигурирования интерфейса)

COS второго уровня	Нет
IOS второго уровня	Нет
IOS 3500	Нет

б) Доверять QoS-информации от другого коммутатора уровня распределения, основного уровня или коммутатора третьего уровня на уровне доступа (*ячейки телефона*).

COS третьего уровня	`set port qos mod/port port-based` `set port qos mod/port trust trust-dscp`

IOS третьего уровня	`no mls qos vlan-based`
	`mls qos trust dscp`
	(режим конфигурирования интерфейса)

COS второго уровня	`set port qos mod/port trust trust-cos`
IOS второго уровня	`no mls qos trust cos` (режим конфигурирования интерфейса)
IOS 3500	Нет

QoS-информации от других коммутаторов в QoS домене можно доверять. При этом предполагается, что *все* коммутаторы в QoS-домене сконфигурированы для последовательного соблюдения правил качества обслуживания.

В этих соединениях качество обслуживания основывается на поэтах, поскольку каждая VLAN сеть, поддерживаемая каналом, имеет собственные, уже изученные и модифицированные QoS-значения. Коммутаторы третьего уровня могут доверять входящей DSCP-информации, однако коммутаторы второго уровня могут доверять только входящим CoS-значениям.

в) Не доверять QoS-информации отправителей, расположенных за пределами QoS домена (*необязательно*).

COS третьего уровня	`set port qos mod/port cos 0` `set port qos mod/port trust untrusted`

IOS третьего уровня	`mls qos cos 0`
	`no mls qos trust`
	(режим конфигурирования интерфейса)

COS второго уровня	`set port qos mod/port cos 0` `set port qos mod/port trust untrusted`

IOS второго уровня	`mls qos cos 0`
	`no mls qos trust`
	(режим конфигурирования интерфейса)

IOS 3500	Нет

Немаркированные фреймы получают CoS-значение 0. Это также приводит к тому, что входящие DSCP-значения с помощью CoS-DSCP-преобразования заменяются на нулевое значение.

2. Регулировка входящего QoS-DSCP-преобразования (*необязательно; только для коммутаторов третьего уровня*).

Система COS	`set qos cos-dscp-map 0 8 16 26 32 46 48 56` `set qos ipprec-dscp-map 0 8 16 26 32 46 48 56`

Система IOS	`mls qos map cos-dscp 0 8 16 26 32 46 48 56`
	`mls qos map ip-prec-dscp 0 8 16 26 32 46 48 56`
	(обе команды вводятся в режиме глобальной конфигурации)

Можно сконфигурировать устройство так, чтобы значение CoS 3 преобразовывалось в код DSCP 26 (AF31), а значение CoS 5 — в код DSCP 46 (EF). Стандартные значения несколько отличаются от общепринятых.

3. Применение QoS-правила к голосовому трафику (*необязательно; только для коммутаторов третьего уровня*)

а) Указание трафика с помощью списков ACL

Система COS	`set qos acl ip` *acl-name* `dscp 26 tcp any any range 2000 2002`

Система IOS	`ip access-list extended` *acl-name*
	(режим глобальной конфигурации)
	`permit tcp any any range 2000 2002 dscp 26`
	`exit`
	(последние команды вводятся в режиме конфигурирования списка доступа)

В данном случае TCP-порты 2000, 2001 и 2002 протокола управления голосом SCCP соответствуют условиям списка. Таким фреймам задается DSCP-значение 26 (AF31), даже если оно уже установлено.

Если используются другие протоколы управления голосом, можно изменить ACL-список для соответствия необходимым номерам портов.

б) Указание QoS-правила (*только для системы IOS третьего уровня*).

Система COS	Нет
Система IOS	`policy-map` *policy-name* (режим глобальной конфигурации) `class` *class-name* `access-group` *acl-name*

В этом правиле используется класс для проверки трафика на ACL-списка.

в) Применение QoS-правила к голосовой VLAN-сети.

Система COS	`commit qos acl acl-name` `set qos acl map acl-name voice-vlan`

Система IOS	`interface vlan voice-vlan`
	(режим глобальной конфигурации)
	`service-policy input policy-name`
	(режим конфигурирования интерфейса)

Можно применить это QoS-правило ко всем портам, поддерживающим голосовую VLAN-сеть. Такой подход является эффективным способом использования механизма QoS в одной определенной VLAN-сети внутри магистрального канала.

6. Конфигурирование планировщика голосовых потоков на выходных портах.

COS третьего уровня	`set port qos mod/port port-based`
	`set qos map 1p2q2t tx 2 1 cos 3`
	`set qos map 2q2t tx 2 1 cos 3`

IOS третьего уровня	`no mls qos vlan-based`
	`wrr-queue cos-map 2 1 3`
	(режим конфигурирования интерфейса)

COS второго уровня	`set qos map 1p2q2t tx 2 1 cos 3` `set qos map 2q2t tx 2 1 cos 3`
IOS второго уровня	`wrr-queue cos-map 2 1 3` (режим конфигурирования интерфейса)
IOS 3500	Нет

Стандартно все фреймы с CoS-значением, равным пяти, отправляются в очередь со строгим приоритетом. Фреймы с CoS-значением, равным трем, отправляются в низкоприоритетную очередь. Таблица планировщика позволяет обеспечить отправку фреймов управления голосом (значение CoS 4) в высокоприоритетную очередь и их первоочередную по отношению к остальному трафику обработку.

Пример конфигурирования QoS для передачи голосовых данных

За информацией о конфигурационных командах обратитесь к примеру конфигурирования параметров качества обслуживания в разделе "13.2: Конфигурирование средств QoS". Там же представлен полный пример конфигурирования голосовых служб, охватывающий различные коммутирующие платформы в многоуровневой конструкции сети.

14.3: голосовые модули

- **Шлюз доступа Catalyst 4000** функционирует в качестве WAN-маршрутизатора, H.323-шлюза в IP-сети (VoIP) и SCCP-группы процессоров цифровых сигналов (Digital Signal Processor – DSP) для программного обеспечения Cisco CallManager. Этот модуль также поддерживает перечисленные ниже интерфейсы.

- Один магистральный 802.1Q-канал Gigabit Ethernet в объединительную плату коммутатора, поддерживающий до шести VLAN-сетей.
- Две *платы интерфейса голос/WAN (Voice/WAN Interface Card — VWIC)* — одно- или двухпортовые магистральные каналы T1 и E1.
- Одна *плата голосового интерфейса (Voice Interface Card — VIC)* — двухпортовая *станция внешнего обмена (Foreign Exchange Station — FXS)*, двухпортовый *офис внешнего обмена (Foreign Exchange Office — FXO)* или двухпортовый ISDN BRI/ST-интерфейс.
- Одно гнездо для *платы WAN-интерфейса (WAN Interface Card — WIC)* — однопортовый блок DSU/CSU 56/64 Кбит/с, двухпортовый последовательный асинхронный/синхронный интерфейс.

- **8-портовый RJ-21 FXS-модуль Catalyst 4000** — VoIP-шлюз, имеющий восемь FXS-интерфейсов для аналоговых телефонов и факсов.
- **Коммутатор шлюза доступа Catalyst 4224** — комбинированное устройство, содержащее 24-портовый Ethernet-коммутатор на 10/100 Мбит/с, восьмипортовый FXS-модуль аналоговой телефонии и 3 гнезда для плат VWIC, VIC и WIC.
- **Восьмипортовый модуль голосовых каналов T1 и служб Catalyst 6000** — VoIP-шлюз к восьми каналам T1 (или E1) ISDN PRI или магистральным каналам CAS.
- **24-портовый FXS-модуль Catalyst 6000** — VoIP-шлюз для двадцати четырех аналоговых станций.

Конфигурирование шлюзов доступа Catalyst 4000 и 4224

1. Открытие CLI-сеанса *(Command-Line Interface — интерфейс командной строки)* с модулем шлюза доступа *(только для Catalyst 4000)*.

Система COS	`session` *module*
Система IOS	Нет

Из интерфейса командной строки модуля Supervisor коммутатора открывается сеанс Telnet-связи с модулем шлюза доступа. В этом модуле для ввода всех конфигурационных команд и отображения конфигурации используется IOS-интерфейс командной строки маршрутизатора.

2. Конфигурирование шлюза доступа.

Совет

Модуль шлюза доступа Catalyst 4000 и коммутатор шлюза доступа Catalyst 4224 используют команды программного обеспечения Cisco IOS для маршрутизаторов. Более подробная информация по конфигурированию приведена в главе 12 "Управление коммутаторами", книги *Cisco Field Manual: Router Configuration* издательства Cisco Press, или на странице *Руководство по конфигурированию голосовых, видео- и факс-приложений в операционной системе Cisco IOS (Cisco IOS Voice, Video, and Fax Configuration Guide)*, расположенной по адресу www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fvvfax_c/index.htm.

В частности шлюз доступа конфигурируется как маршрутизатор и H.323 VoIP-шлюз. В главе 12, "Управление коммутаторами", приведена необходимая информация по настройке голосовых портов, точек вызова (dial peers), H.323-шлюзов и QoS-функций, необходимых для доставки голосовых данных. Также в указанной главе описана SRS-телефония (*Survivable Remote Site — надежный удаленный участок*).

Восьмигортовый FXS-модуль Catalyst 4000 необходимо настраивать в сеансе конфигурирования модуля шлюза доступа. FXS-модуль не имеет собственного сеансового интерфейса, поэтому его невозможно настроить отдельно от модуля шлюза доступа.

Конфигурирование голосовых модулей Catalyst 6000

1. Идентификация конфигурационного сервера.

   а) Использование DHCP-сервера (*необязательно*).

Система COS	`set port voice interface` *mod/port* `dhcp enable [vlan` *vlan*`]`
Система IOS	Нет

Для получения своего IP-адреса модуль отправляет DHCP-запрос. Кроме того, DHCP-сервер должен вернуть адреса TFTP-сервера и DNS-сервера, а также адрес стандартного шлюза. Каждый порт в восьмипортовом T1/E1-модуле должен получить уникальный IP-адрес. Для 24-портового FXS-модуля необходим только один IP-адрес.

Чтобы назначить порт голосового шлюза определенной голосовой VLAN-сети, можно использовать ключевое слово `vlan` и параметр `vlan` (номер сети от 1 до 1000 или от 1025 до 4094).

Совет

Для просмотра назначенных каждому порту или модулю голосового шлюза MAC-адресов используется команда `show port voice`.

   б) Использование статической конфигурации (без DHCP-сервера) (*необязательно*).

Система COS	`set port voice interface` *mod/port* `dhcp disable` *ipaddrspec* `tftp` *ipaddr* `[vlan` *vlan*`] [gateway` *ipaddr*`] [dns` *[ipaddr] [domain-name]*`]`
Система IOS	Нет

В ситуации, когда использование DHCP-сервера нецелесообразно или невозможно, допускается присвоение каждому порту голосового шлюза статических конфигурационных значений. Поле *ipaddrspec* задает IP-адрес и маску в формате *ip-address/mask* (маска в точечной форме записи) или *ip-address/dice* (количество битов маски от 1 до 31), или *ip-address* (значение маски происходит из класса сети). Каждый порт в восьмипортовом T1/E1-модуле должен получить уникальный IP-адрес. Для 24-портового FXS-модуля необходим только один IP-адрес.

Ключевое слово `tftp` определяет адрес TFTP-сервера (*ipaddr*). Порт голосового модуля можно назначить определенной голосовой VLAN-сети, используя ключевое слово `vlan`. Стандартный шлюз может быть задан с помощью ключевого слова `gateway`. Можно использовать ключевое слово `dns` для идентификации DNS-сервера по IP-адресу (*ipaddr*) и доменному имени (*domain-name* — текстовая строка). Если какой-либо из параметров DNS опущен, используются DNS-значения от модуля Supervisor коммутатора. Определить их можно с помощью команд `set ip dns server` и `set ip dns domain`.

2. Конфигурирование голосового шлюза в CallManager Cisco.

   После того как модуль получает свой IP-адрес и серверную информацию, он может загрузить остальные конфигурационные параметры. Адреса серверов управления вызовами (CallManager) Cisco предоставляются TFTP-серверам. Серверы CallManager должны быть сконфигурированы так, чтобы они могли контролировать все аспекты работы модуля.

Конфигурация функции

Восьмипортовый T1-модуль Catalyst 6000 расположен в гнезде 2. Он настроен на использование DHCP-сервера для портов 2/7 и 2/8. Портам с 1 по 6 статически назначаются IP-адреса, маски, а также адрес TFTP-сервера 192.168.3.200 и адрес стандартного шлюза 192.168.217.1. Все 8 портов сконфигурированы на использование в качестве голосовой VLAN сети VLAN 21. Следует заметить, что каждому T1-порту требуется уникальный IP-адрес.

В гнездах 3 и 4 расположены два 24-портовых FXS-модуля Catalyst 6000. Для получения конфигурационных сведений модуль в гнезде 3 использует DHCP-сервер. Модуль в гнезде 4 конфигурируется статически. Всем двадцати четырем портам этих модулей требуется только один IP-адрес.

Система COS

```
set port voice interface 2/1 dhcp disable
192.168.217.10 255.255.255.0 vlan 21 tftp 192.168.3.200
gateway 192.168.217.1
set port voice interface 2/2 dhcp disable
192.168.217.11 255.255.255.0 vlan 21 tftp 192.168.3.200
gateway 192.168.217.1
set port voice interface 2/3 dhcp disable
192.168.217.12 255.255.255.0 vlan 21 tftp 192.168.3.200
gateway 192.168.217.1
set port voice interface 2/4 dhcp disable
192.168.217.13 255.255.255.0 vlan 21 tftp 192.168.3.200
gateway 192.168.217.1
set port voice interface 2/5 dhcp disable
192.168.217.14 255.255.255.0 vlan 21 tftp 192.168.3.200
gateway 192.168.217.1
set port voice interface 2/6 dhcp disable
192.168.217.15 255.255.255.0 vlan 21 tftp 192.168.3.200
gateway 192.168.217.1
set port voice interface 2/7 dhcp enable vlan 21
set port voice interface 2/8 dhcp enable vlan 21
```

```
set port voice interface 3 dhcp enable vlan 21
set port voice interface 4 dhcp disable
192.168.217.30/24 vlan 21 tftp 192.168.3.200 gateway
192.168.217.1
```

Система IOS	Нет

Отображение информации о голосовых модулях

В табл. 14.3 перечислены некоторые команды коммутатора, которые можно использовать для получения полезной информации о модулях голосового шлюза Catalyst 6000.

Таблица 14.3. Команды коммутатора для отображения информации о модуле голосового шлюза Catalyst

Функция отображения	Операционная система коммутатора	Команда
Конфигурация голосовых портов и сервера	COS	`show port voice interface [mod[/port]]`
	IOS	Нет
Состояние CallManager T1/E1 и DSP	COS	`show port [mod[/port]]`
	IOS	Нет
Состояние трубки и линии FXS-порта	COS	`show port [mod[/port]]`
	IOS	Нет
Статистика ошибок голосового порта FDL (Facilities Data Link)	COS	`show port voice fdl [mod[/port]]`
	IOS	Нет
Информация по активному вызову порта T1/E1	COS	`show port voice active [mod[/port]] [all | call | conference | transcoder] [ipaddr]`
	IOS	Нет
Информация по активному вызову FXS-порта	COS	`show port voice active [mod[/port]] call`
	IOS	Нет

Дополнительная литература

Рекомендуемые ниже источники предоставляют дополнительную информацию по темам, рассматриваемым в этой главе.

Руководства по проектированию систем IP-телефонии Cisco

Руководство по проектированию сети IP-телефонии Cisco (Cisco IP Telephony Network Design Guide): www.cisco.com/univercd/cc/td/doc/product/voice/ip_tele/network/index.htm.

Руководство по расчету параметров качества обслуживания в IP-телефонии Cisco (Cisco IP Telephony QoS Design Guide): www.cisco.com/univercd/cc/td/doc/product/voice/ip_tele/avvidqos/

Руководство по решениям в области IP-телефонии (IP Telephony Solution Guide): www.cisco.com/univercd/cc/td/doc/product/voice/ip_tele/solution/index.htm.

Книги по IP-телефонии Cisco

David Lovell, *Cisco IP Telephony*. Cisco Press.

Jonathan Davidson, *Deploying Cisco Voice over IP Solutions*, Cisco Press

Scott Keagy, *Integrating Voice and Data Networks*, Cisco Press.

Deal, Nelson, and Smith. *Developing Cisco IP Phone Services*. Cisco Press.

Smith, Alexander, Pearce, and Whetten, *Cisco CallManager Fundamentals*, Cisco Press.

Paul Giralt, Addis Hallmark, *Troubleshooting Cisco IP Telephony*. Cisco Press.

Линейное питание

DTE-питание через MDI в спецификации IEEE P802.3af (IEEE P802.3af DTE Power via MDI): http://grouper.ieee.org/groups/802/3/af/.

Протоколы передачи голосовых данных

Протокол создания сеанса (Session Initiation Protocol, SIP), RFC 2543: ftp://ftp.isi.edu/in-notes/rfc2543.txt.

ITU-стандарт H.323: http://www.itu.int/home/index.html

Протокол управления мультимедийным шлюзом (Multimedia Gateway Control Protocol, MGCP), версия 1.0, RFC 2705: ftp://ftp.isi.edu/in-notes/rfc2705.txt.

Протокол Megaco: http://www.ietf.org/html.charters/megaco-charter.html, а также www.ietf.org/rfc/rfc3015.txt.

Транспортный протокол реального времени (Real-Time Transport Protocol — RTP), RFC 1889: www.cs.columbia.edu/~hgs/rtp/.

Качество обслуживания при передаче голосовых данных

CIM Voice Internetworking: VoIP Quality of Service, Cisco Systems.

[redacted]

Приложение A

Краткий справочник по кабельным системам

В табл. A.1 перечислены различные типы *конвертеров гигабитного интерфейса (Gigabit Interface Converters — GBIC)*, которые можно использовать в коммутаторах Cisco Catalyst.

Таблица A.1. GBIC-конвертеры для коммутаторов Cisco Catalyst

Тип GBIC-конвертера	Номер в классификаторе Cisco (Cisco Part Number)	Разъем	Тип передающей среды
1000BASE-T	WS-G5482	RJ-45	Витая пара UTP категории 5
Gigastack	WS-X3500-XL	Стековый кабель (Stack cable)	Стековый кабель
	CAB-GS-50CM		Стековый кабель длиной 50 см
	CAB-GS-1M		Стековый кабель длиной 1 м
1000BASE-SX	WS-G5484	SC	Многомодовый волоконно-оптический кабель
1000BASE-LX/LH	WS-G5485	SC	Одномодовый или многомодовый волоконно-оптический кабель
	CAB-GELX-625	SC	Соединительный кабель согласования (mode conditioning patch cable)
1000BASE-ZX	WS-G5487	SC	Одномодовый волоконно-оптический кабель

На сетевые кабельные системы всегда накладываются ограничения по протяженности, которые зависят от используемой передающей среды и поддерживаемой полосы пропускания. В табл. A.2 приводятся краткие сведения о максимальной длине кабеля для различных сетевых сред и типов кабеля.

Таблица А.2. Длина кабеля для сетевых сред и кабельных систем

Передающая среда	Тип кабеля	Максимальная длина (м, в скобках — футов)
10/100BASE-TX Ethernet	Кабель UTP[1] категории 5	100 (328)
100BASE-FX	MMF 62.5/125	400 (полудуплексный)
		2000 (дуплексный)
	SMF	10000
1000BASE-CX	STP	25 (82)
1000BASE-T	EIA/TIA UTP, категория 5 (4 пары)	100 (328)
1000BASE-SX	MMF 62.5 мкм, 160 МГц/км	220 (722)
	MMF 62.5 мкм, 200 МГц/км	275 (902)
	MMF 50.0 мкм, 400 МГц/км	500 (1640)
	MMF 50.0 мкм, 500 МГц/км	550 (1804)
1000BASE-LX/LH[2]	MMF 62.5 мкм, 500 МГц/км	550 (1804)
	MMF 50.0 мкм, 400 МГц/км	550 (1804)
	MMF 50.0 мкм, 500 МГц/км	550 (1804)
	SMF 9/10	10000 (32 810)
1000BASE-ZX	SMF	от 70 до 100 (кабельный модем)
SONET	MMF (62.5 или 50.0 мкм)	9000 (1.5 мили)
	SMI[3]	15000 (9 миль)
	Одномодовый Большой протяженности (Single-mode long reach)	45000 (28 миль)
FDDI	MMF	2000 (1.2 мили)
	SMF	15000 (9.3 мили)
Token Ring (IEEE 802.5)	STP	500 (1640)
Token Ring (IEEE 802.5)	EIA/TIA категория UTP 5	100 (328)
ISDN BRI	UTP, RJ-45	10 (32,8)
Асинхронный EIA/TIA-232	2400 бод	60 (200)
	4800 бод	30 (100)
	9600 бод	15 (50)
	19200 бод	15 (50)
	38400 бод	15 (50)
	57600 бод	7.6 (25)

[1] *UTP — неэкранированная витая пара, STP — экранированная витая пара, MMF — многомодовое оптоволокно, SMF — одномодовое оптоволокно. — Прим. ред.*

[2] *При использовании GBIC-конвертеров 1000BASE-LX/LH с многомодовым волокном 62,5 мкм на расстояниях более 300 м (984 футов) необходимо использовать соединительный кабель состояния. Сведения по установке и применению приведены на Web-странице* `www.cisco.com/univercd/cc/td/doc/product/lan/cat5000/cnfg_nts/ethernet/5421_01.htm`. *— Прим. ред.*

[3] *SMI — одномодовый кабель средней протяженности. — Прим. ред.*

Окончание табл. А.2

Передающая среда	Тип кабеля	Максимальная длина (м, в скобках — футов)
	115200 бод	3,7 (12)
Синхронный EIA/TIA-449 со сбалансированными драйверами, включая X.21 и V.35 (Sync EIA/TIA-449 with balanced drivers, including X.21 and V.35)	2400 бод	1250 (4100)
	4800 бод	625 (2050)
	9600 бод	312 (1025)
	19200 бод	156 (513)
	38400 бод	78 (256)
	56000 бод	31 (102)
	T1 (1.544 Мбит/с)	15 (50)

Часто выясняется, что необходимо определить разводку проводов в соединениях для различных типов кабелей. Разъем RJ-45 широко применяется во многих средах, но с разной разводкой для каждой из них. В табл. А.3 демонстрируется разводка проводов для разъема RJ-45 при использовании в определенных передающих средах.

Прямые лабораторные соединения устройств

Для лабораторных исследований или в определенных условиях может понадобиться соединить два коммутатора или два маршрутизатора напрямую (back-to-back). Обычно для соединения интерфейсов маршрутизаторов используется какое-либо другое активное устройство. Например, активная роль в соединении маршрутизаторов отводится таким элементам, как Ethernet-концентратору или коммутатору, *модулю подключения к среде (Media Attachment Unit — MAU)* Token Ring, а также *открытой коммутируемой телефонной сети (Public Switched Telephone Network — PSTN)*. Если такие элементы недоступны, например, в лабораторной среде, то для прямого соединения необходим специальный кабель.

Внимание!

Два интерфейса Token Ring соединить таким способом невозможно. Для соединений Token Ring требуется какое-либо активное устройство, как, например, MAU и Token Ring-коммутатор, терминирующий соединение.

Ethernet-соединения

Обычно *плата сетевого интерфейса (Network Interface Card — NIC)* узла 10BASE-T или 10/100BASE-TX подключается к коммутатору посредством *прямого (straight-through)* кабеля пятой категории UTP. При этом контакты 1 и 2 разъема RJ-45 формируют одну пару, а контакты 3 и 6 — другую. Для непосредственного соединения портов двух коммутаторов требуется *перекрестный (crossover)* кабель.

Таблица А.3. Распайка проводов для разъема RJ-45 в зависимости от типа передающей среды

Контакт в разъеме RJ-45	Консоль маршрутизатора (DTE)	Ethernet UTP	Token Ring UTP	ISDN BRI S/T TE	ISDN BRI U	CT1/PRI CSU	CE1/ PRI	56/64 Кбит/с DSU/CSU[1]	T1/E1[1]	
		10/100	1000							
1	RTS	TX+	TP0+	GND			Rcv Ring	TX Tip	TX Ring	RX
2	DTR	TX–	TP0–	GND			Rcv Tip	TX Ring	TX Tip	RX
3	TxD	RX–	TP1+	TX+	TX+			TX Shld		
4	GND		TP2+	RX+	RX+	Tip or Ring	Ring	RX Tip		TX
5	GND		TP2–	RX–	RX–	Tip or Ring	Tip	RX Ring		TX
6	RxD	RX–	TP1–	TX–	TX–			RX Shld		
7	DSR		TP3+	GND					RX Tip	
8	CTS		TP3						RX Ring	

Таблица А.4. Распайка проводов разъема RJ-45 для перекрестных кабелей

Контакт разъема RJ-45 Конец А	Описание конца А	Описание конца В	Контакт разъема RJ-45 Конец В
1	TX+	RX+	3
2	TX–	RX–	6
3	RX+	TX+	1
4			4
5			5
6	RX–	TX–	2
7			7
8			8

[1] *Фактически в данном случае используется разъем RJ-48. — Прим. ред.*

С помощью перекрестного кабеля соединяется пара контактов 1 и 2 на одном конце и пара контактов 3 и 6 — на другом. Аналогично контакты 3 и 6 подключены к контактам 1 и 2. В табл. А.4 приведена распайка соединений для обоих концов RJ-45 перекрестного кабеля.

Асинхронные последовательные соединения

Для асинхронного последовательного соединения, такого, как дополнительный порт (Aux) и линия на сервере доступа, требуется соединение типа RJ-45. Для непосредственного лабораторного соединения двух асинхронных портов маршрутизаторов требуется использовать *скрученный*[1] *(rollover)* кабель. Скрученные кабели обычно представляют собой плоские восьмипроводниковые кабели с разъемами RJ-45, в которых контакт 1 на одном конце соединяется с контактом 8 на другом, контакт 2 соединяется с контактом 7 и так далее. Корпорации Cisco обычно поставляет необходимый кабель в составе набора консольных проводов. В табл. А.5 представлена разводка проводов для обоих концов скрученного кабеля.

Таблица А.5. Распайка проводов разъема RJ-45 для консольных кабелей

Контакт разъема RJ-45 Конец А	Описание конца А	Описание конца В	Контакт разъема RJ-45 Конец В
1	RTS	CTS	8
2	DTR	DSR	7
3	TxD	RxD	6
4	GND	GND	5
5	GND	GND	4
6	RxD	TxD	3
7	DSR	DTR	2
8	CTS	RTS	1

CSU/DSU-соединения на 56/64 Кбит/с

Обычно, если в маршрутизаторе имеется интегрированный или внешний CSU/DSU-блок для последовательного интерфейса на 56/64 Кбит/с, он подключается к конечному оборудованию провайдера. Провайдер служб или PSTN-сеть устанавливает активный канал между двумя блоками CSU/DSU и маршрутизаторами.

Лабораторные последовательные соединения обычно можно создавать между последовательными интерфейсами двух маршрутизаторов с помощью одного последовательного кабеля для *оконечного оборудования (Data Terminating Equipment — DTE)* и одного последовательного кабеля для *коммуникационного оборудования (Data Communication Equipment — DCE)*. Один маршрутизатор становится DTE-устройством, а другой — DCE-устройством и должен обеспечивать синхронизацию. Однако если имеются два маршрутизатора с интегрированными CSU/DSU-блоками, то не существует способ получить доступ к последовательному физическому интерфейсу. В таком случае лабораторный кабель 56/64 можно по-

[1] *Чаще всего такой кабель называется консольным. Прим. ред.*

лучить за счет перекрещивания передающих и принимающих пар. В табл. А.6 показаны соединения контактов обоих концов RJ-48 (и RJ-45) кабеля.

Таблица А.6. Распайка проводов разъема RJ-48 для лабораторных CSU/DSU-соединений на скорости 56/54 Кбит/с

Контакт разъема RJ-48 Конец А	Описание конца А	Описание конца В	Контакт разъема RJ-48 Конец В
1	TX Ring	RX Ring	7
2	TX Tip	RX Tip	8
3			3
4			4
5			5
6			6
7	RX Tip	TX Tip	1
8	RX Ring	TX Ring	2

CSU/DSU-соединения T1/E1

Кроме того, при помощи специально изготовленного кабеля можно организовать лабораторное соединение между двумя маршрутизаторами с интегрированными CSU/DSU-блоками T1/E1. В таком кабеле также необходимо перекрестно соединить передающие и принимающие пары. В табл. А.7 приведена распайка проводов обоих концов RJ-48 (и RJ-45) кабеля.

Таблица А.7. Распайка проводов разъема RJ-48 для лабораторных CSU/DSU-соединений T1/E1

Контакт разъема RJ-48 (А)	Описание разъема А	Описание разъема В	Контакт разъема RJ-48 (В)
1	RX (вход)	TX (вход)	4
2	RX (вход)	TX (вход)	5
3			3
4	TX (выход)	RX (выход)	1
5	TX (выход)	RX (выход)	2
6			6
7			7
8			8

В этом приложении...

- **Б.1: номера IP-протоколов.**
- **Б.2: ICMP-типы и коды.**
- **Б.3: стандартные номера IP-портов.**
- **Б.4: стандартные IP-адреса многоадресатного вещания.**
- **Б.5: коды Ethernet.**

Приложение Б

Номера портов, протоколов и другие стандартные номера

Б.1: номера IP-протоколов

Любой протокол более высокого уровня внутри пакета IPv4 идентифицируется с помощью 8-битового поля, которое называется *Protocol* (протокол). Формат заголовка пакета IPv4 представлен на рис. Б.1, поле *Protocol* выделено. На рис. Б.2 демонстрируется формат заголовка пакета шестой версии протокола IP (v6), в котором номер протокола содержится в заменяющем поле Next Header (следующий заголовок).

Стандартные (well-known), или назначенные IP-протоколы, регистрируются в *Агентстве по выделению имен и уникальных параметров протоколов Internet (Internet Assigned Numbers Authority — IANA)*. Сведения, приведенные в этом приложении, воспроизводятся с разрешения организации. Наиболее актуальная информация, касающаяся назначений номеров IP-протоколов, публикуется на Web-странице www.iana.org/numbers.htm (ссылка "Protocol Numbers" — номера протоколов).

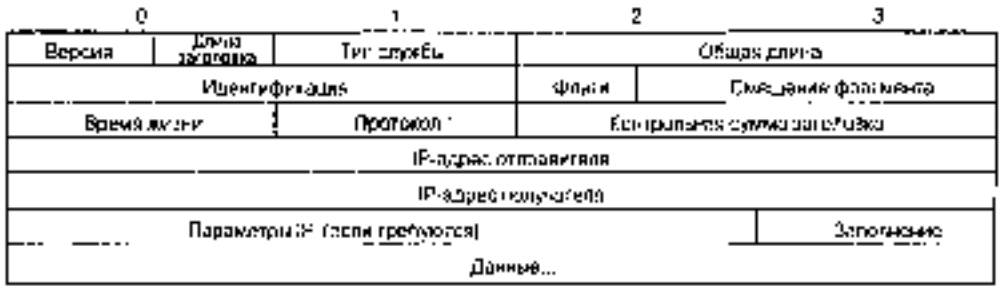


Рис. Б.1. Формат заголовка IPv4 с полем Protocol

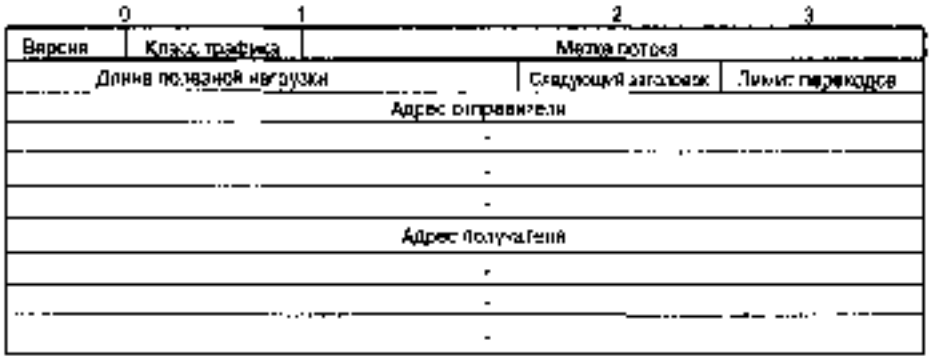


Рис. Б.2. Формат заголовка IPv6 с полем Next Header

В табл. Б.1 представлены зарегистрированные номера IP-протоколов наряду с ключевыми словами (или аббревиатурами протоколов), полные названия протоколов, а также номера документов RFC (если имеются).

Таблица Б.1. Зарегистрированные номера IP-протоколов, ключевые слова, названия и связанные с ними RFC-документы

Ключевое слово, протокол, ссылки	Номер
HOPOPT Вариант протокола IPv6 для транзитной передачи (IPv6 Hop-by-Hop Option) RFC 1883	0
ICMP Протокол управляющих сообщений Internet (Internet Control Message) RFC 792	1
IGMP Межсетевой протокол управления группами (Internet Group Management) RFC 1112	2
GGP Межшлюзовый протокол (Gateway-to-Gateway) RFC 823	3
IP IP в IP (инкапсуляция) RFC 2003	4
ST Поток (Stream) RFC 1190, RFC 1819	5
TCP Протокол управления передачей (Transmission Control) RFC 793	6
CBT CBT	7
EGP Протокол внешнего шлюза (Exterior Gateway Protocol) RFC 888	8

Продолжение табл. Б.1

Ключевое слово, протокол, ссылки	Номер
IGP Протокол внутреннего шлюза (любой частный внутренний шлюз) IANA (используется Cisco для IGRP)	9
BBN-RCC-MON Мониторинг BBN RCC	10
NVP-II Протокол сетевой передачи голоса (Network Voice Protocol) RFC 741	11
PUP Универсальный протокол PARC (PARC Universal Protocol)	12
ARGUS ARGUS	13
EMCON EMCON	14
XNET Сетевой отладчик (Cross Net Debugger)	15
CHAOS Chaos	16
UDP Протокол пользовательских дейтаграмм (User Datagram) RFC 768	17
MUX Мультиплексирование (Multiplexing)	18
DCN-MEAS Подсистемы измерения DCN (DCN Measurement Subsystems)	19
HMP Протокол мониторинга узла (Host Monitoring) RFC 869	20
PRM Радиоизмерения пакетов (Packet Radio Measurement)	21
XNS-IDP IDP в сетях XEROX (XEROX NS IDP)	22
TRUNK-1 Trunk-1	23
TRUNK-2 Trunk-2	24
LEAF-1 Leaf-1	25
LEAF-2 Leaf-2	26
RDP Протокол достоверных данных (Reliable Data Protocol) RFC 908	27

Ключевое слово, протокол, ссылки	Номер
IRTP Протокол надежных транзакций Internet (Internet Reliable Transaction) RFC 938	28
ISO-TP4 Транспортный протокол ISO класса 4 (ISO Transport Protocol Class 4) RFC 905	29
NETBLT Протокол массовой передачи данных (Bulk Data Transfer Protocol) RFC 969	30
MFE-NSP Протокол сетевых служб MFE (MFE Network Services Protocol)	31
MERIT-INP Межузловой протокол MERIT (MERIT Internode Protocol)	32
SEP Протокол последовательного обмена (Sequential Exchange Protocol)	33
3PC Протокол подключения посредника (Third Party Connect Protocol)	34
IDPR Протокол междоменной маршрутизации политик (Inter-Domain Policy Routing Protocol) RFC 1479	35
XTP XTP	36
DDP Протокол доставки дейтаграмм (Datagram Delivery Protocol)	37
IDPR-CMTP Транспортный протокол управляющих сообщений IDPR (IDPR Control Message Transport Protocol)	38
TP++ Транспортный протокол TP++	39
IL Транспортный протокол IL	40
IPv6 Протокол IP версии 6	41
SDRP Протокол маршрутизации по запросу отправителя (Source Demand Routing Protocol)	42
IPv6-Route Маршрутный заголовок для IPv6 (Routing Header for IPv6)	43
IPv6-Frag Заголовок фрагмента для IPv6 (Fragment Header for IPv6)	44
IDRP Протокол междоменной маршрутизации (Inter-Domain Routing Protocol)	45
RSVP Протокол резервирования ресурсов (Resource ReSerVation Protocol) RFC 2205	46

Ключевое слово, протокол, ссылки	Номер
GRE Общая инкапсуляция для маршрутизации (General Routing Encapsulation) RFC 1701	47
MHRP Протокол маршрутизации для мобильных узлов (Mobile Host Routing Protocol)	48
BNA BNA	49
ESP Инкапсулированная безопасная полезная нагрузка (Encapsulating Security Payload) RFC 2406	50
AH Заголовок аутентификации (Authentication Header) RFC 2402	51
I-NLSP Интегрированная функция безопасности сетевого уровня (Integrated Net Layer Security) TUBA	52
SWIPE Протокол IP с шифрованием (IP with Encryption)	53
NARP Протокол преобразования адресов NBMA (NBMA Address Resolution Protocol) RFC 1735	54
MOBILE Поддержка мобильности в IP (IP Mobility) RFC 2002	55
TLSP Протокол безопасности транспортного уровня (Transport Layer Security Protocol) с использованием администрирования Kryptonet-ключей	56
SKIP SKIP	57
IPv6-ICMP Протокол ICMP для IPv6 RFC 2463	58
IPv6-NoNxt Без поля Next Header для IPv6 RFC 2460	59
IPv6-Opts Параметры получателей для IPv6 (Destination Options for IPv6) RFC 2460	60
Любой внутренний протокол узла IANA	61
CFTP CFTP	62
Любая локальная сеть IANA	63

Ключевое слово, протокол, ссылки	Номер
SAT-EXPAK SATNET and Backroom EXPAK	64
KRYPTOLAN Kryptolan	65
RVD Протокол удаленного виртуального диска, разработанный в MIT (MIT Remote Virtual Disk Protocol)	66
IPPC Пакетная основа Internet Pluribus (Internet Pluribus Packet Core)	67
Любая распределенная файловая система IANA	68
SAT-MON Мониторинг SATNET	69
VISA Протокол VISA	70
IPCV Утилиты пакетной основы Internet (Internet Packet Core Utility)	71
CPNX Компьютерная протокольная сеть (Computer Protocol Network Executive)	72
CPHB Компьютерный тестовый протокол (Computer Protocol Heart Beat)	73
WSN Промежуточная сеть Wang (Wang Span Network)	74
PVP Протокол пакетного видео (Packet Video Protocol)	75
BR-SAT-MON Мониторинг Backroom SATNET (Backroom SATNET Monitoring)	76
SUN-ND SUN ND PROTOCOL-Temporary	77
WB-MON WIDEBAND-мониторинг	78
WB-EXPAK WIDEBAND EXPAK	79
ISO-IP Internet-протокол ISO	80
VMTP VMTP RFC 1045	81
SECURE-VMTP SECURE-VMTP	82
VINES Виртуальная интегрированная сетевая служба	83
TTP TTP	84
NSFNET-IGP NSFNET-IGP	85

Продолжение табл. Б.1

Ключевое слово, протокол, ссылки	Номер
OGP Несходный шлюзовый протокол (Dissimilar Gateway Protocol)	66
TCF TCF	67
EIGRP EIGRP CISCO	88
OSPFIGP OSPFIGP RFC 2328	89
Sprite-RPC Протокол Sprite RPC	90
LARP Протокол преобразования адресов Locus (Locus Address Resolution Protocol)	91
MTP Протокол многоадресатной транспортировки (Multicast Transport Protocol)	92
AX.25 Фреймы AX.25	93
IPIP Протокол инкапсуляции IP внутри IP (IP-within-IP Encapsulation Protocol) RFC 1853	94
MICP Управляющий протокол мобильных сетей (Mobile Internetworking Control Protocol)	95
SCC-SP Semaphore Communications Sec. Pro.	96
ETHERIP Инкапсуляция Ethernet внутри IP (Ethernet-within-IP Encapsulation)	97
ENCAP Заголовок инкапсуляции (Encapsulation Header) RFC 1241	98
Любая частная схема шифрования IANA	99
GMTP GMTP	100
IFMP Протокол управления потоками Ipsilon (Ipsilon Flow Management Protocol)	101
PNNI PNNI в IP-сетях (PNNI over IP)	102
PIM Не зависящая от протокола многоадресатная передача (Protocol Independent Multicast) RFC 2362 (разреженный режим)	103
ARIS ARIS	104

Ключевое слово, протокол, ссылки	Номер
SCPS SCPS	105
QNX QNX	106
A/N Активные сети (Active Networks)	107
IPComp Протокол сжатия полезной нагрузки IP-пакетов (IP Payload Compression Protocol) RFC 2393	108
SNP Протокол сетей Sitara (Sitara Networks Protocol)	109
Compaq-Peer Протокол равноправных узлов Compaq (Compaq Peer Protocol)	110
IPX-in-IP IPX в IP RFC 1234	111
VRRP Протокол избыточности виртуальных маршрутизаторов (Virtual Router Redundancy Protocol) RFC 2338	112
PGM Протокол надежной транспортировки PGM (PGM Reliable Transport Protocol)	113
Любой протокол без транзитных переходов IANA	114
L2TP Протокол туннелирования второго уровня (Layer Two Tunneling Protocol) RFC 2661	115
DDX Обмен данными D-II (D-II Data Exchange)	116
IATP Протокол интерактивной передачи агента (Interactive Agent Transfer Protocol)	117
STP Протокол доставки по расписанию (Schedule Transfer Protocol)	118
SRP SpectraLink-радио протокол (Spectralink Radio Protocol)	119
UTI UTI	120
SMP Протокол простых сообщений (Simple Message Protocol)	121
SM SM	122
PTP Performance Transparency Protocol	123
ISIS в сетях IPv4	124
FIRE	125

Ключевое слово, протокол, ссылки	Номер
CRTP Транспортный протокол Combat Radio (Combat Radio Transport Protocol)	126
CRUDP Протокол пользовательских дейтаграмм Combat Radio (Combat Radio User Datagram)	127
SSCOPMCE	128
IPLT	129
SPS Безопасная пакетная оболочка (Secure Packet Shield)	130
PIPE Частная IP-инкапсуляция внутри IP (Private IP Encapsulation within IP)	131
SCTP Протокол передачи потокового управления (Stream Control Transmission Protocol)	132
FC Стандарт Fibre Channel	133
Не определены IANA	с 134 по 254
Зарезервирован IANA	255

Б.2: ICMP-типы и коды

С помощью *протокола управляющих сообщений Internet (Internet Control Message Protocol — ICMP)* между маршрутизаторами и другими устройствами передаются сообщения об ошибках, а также управляющие сообщения. ICMP-сообщение инкапсулируется в IP-пакет как полезная нагрузка. На рис. Б.1 показан формат ICMP-сообщения. Следует заметить, что при возникновении ошибки в ICMP-сообщение включаются первые 8 байтов (64 бита) исходной дейтаграммы, вызвавшей ошибку. Эти данные предоставляют информацию о протоколе и номерах портов исходного сообщения, что упрощает поиск и устранение неисправностей.

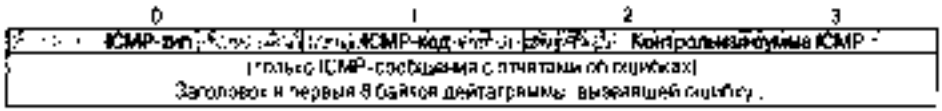


Рис. Б.1. Формат ICMP-сообщения

Коды ICMP регистрируются в организации IANA. Информация, представленная здесь, воспроизведена с разрешения организации. Для получения наиболее актуальных сведений о зарезервированных кодах ICMP рекомендуется обратиться к Web-странице www.iana.org/numbers.html, ссылка "ICMP Type".

В табл. Б.2 перечислены зарезервированные номера протокола ICMP, ICMP-коды (если они применимы), дано их краткое описание, а также ссылки на документы RFC.

Таблица Б.2. Коды ICMP-сообщений

Тип	Код	Название	Ссылка
0		Эхо-ответ (Echo Reply)	RFC 792
1		Не определено	
2		Не определено	
3		Получатель недостижим (Destination Unreachable)	RFC 792
	0	Сеть недостижима (Net Unreachable)	
	1	Узел недостижим (Host Unreachable)	
	2	Протокол недостижим (Protocol Unreachable)	
	3	Порт недостижим (Port Unreachable)	
	4	Требуется фрагментация, но установлен бит "не фрагментировать" (Fragmentation Needed and Don't Fragment Was Set)	
	5	Ошибка маршрута отправителя (Source Route Failed)	
	6	Не известна сеть получателя (Destination Network Unknown)	
	7	Не известен узел-получатель (Destination Host Unknown)	
	8	Отправитель изолирован (Source Host Isolated)	
	9	Сеть получателя административно запрещена (Destination Network Is Administratively Prohibited)	
	10	Узел-получатель административно запрещен (Destination Host Is Administratively Prohibited)	
	11	Сеть получателя недоступна для данного типа обслуживания (Destination Network Unreachable for Type of Service)	
	12	Получатель недоступен для данного типа обслуживания (Destination Host Unreachable for Type of Service)	
	13	Связь административно запрещена (Communication Administratively Prohibited)	RFC 1812
	14	Нарушение приоритета узла (Host Precedence Violation)	RFC 1812
	15	Действует ограничение приоритета (Precedence Cutoff in Effect)	RFC 1812
4		Подавление отправителя (Source Quench)	RFC 792
5		Перенаправление (Redirect)	RFC 792
	0	Перенаправить дейтаграмму сети или подсети (Redirect Datagram for the Network (or Subnet))	
	1	Перенаправить дейтаграмму узлу (Redirect Datagram for the Host)	
	2	Перенаправить дейтаграмму для типа обслуживания и сети (Redirect Datagram for the Type of Service and Network)	
	3	Перенаправить дейтаграмму для заданного типа обслуживания и узла (Redirect Datagram for the Type of Service and Host)	
6		Альтернативный адрес узла (Alternate Host Address)	
	0	Альтернативный адрес для узла (Alternate Address for Host)	
7		Не определено	
8		Эхо-запрос (Echo)	RFC 792
9		Анонс маршрутизатора (Router Advertisement)	RFC 1256

Окончание табл. Б.2

Тип	Код	Название	Ссылка
10		Запрос маршрутизатора (Router Solicitation)	RFC 1256
11		Время истекло (Time Exceeded)	RFC 792
	0	Тот транспортировке истекло время жизни (Time to Live Exceeded in Transit)	
	1	Превышено время сборки фрагментов (Fragment Reassembly Time Exceeded)	
12		Ошибка параметра (Parameter Problem)	RFC 792
	0	Указатель сообщает об ошибке (Pointer Indicates the Error)	
	1	Отсутствует обязательный параметр (Missing a Required Option)	RFC 1108
	2	Неверная длина (Bad Length)	
13		Временная метка (Timestamp)	RFC 792
14		Ответ на временную метку (Timestamp Reply)	RFC 792
15		Информационный запрос (Information Request)	RFC 792
16		Ответ на информационный запрос (Information Reply)	RFC 792
17		Запрос адресной маски (Address Mask Request)	RFC 950
18		Ответ на запрос адресной маски (Address Mask Reply)	RFC 950
19		Зарезервировано (в целях безопасности)	
с 20 по 29		Зарезервированы для экспериментов по проверке надежности	
30		Трассировка маршрута (Traceroute)	RFC 1393
31		Ошибка преобразования дейтаграммы (Datagram Conversion Error)	RFC 1475
32		Перенаправление мобильного узла (Mobile Host Redirect)	
33		Запрос адреса IPv6 (IPv6 Where-Are-You)	
34		Ответ с адресом IPv6 (IPv6 I-Am-Here)	
35		Запрос на мобильную регистрацию (Mobile Registration Request)	
36		Ответ на запрос мобильной регистрации (Mobile Registration Reply)	
37		Запрос доменного имени (Domain Name Request)	
38		Ответ на запрос о доменном имени (Domain Name Reply)	
39		SKIP	
40		Photuris	
	0	Зарезервировано	
	1	Неизвестный индекс параметров безопасности (Unknown Security Parameters Index)	
	2	Параметры безопасности достоверные, но произошла ошибка аутентификации (Valid Security Parameters, but Authentication Failed)	
	3	Параметры безопасности достоверные, но произошел сбой при дешифровке (Valid Security Parameters, but Decryption Failed)	
с 41 по 255		Зарезервированы	

Б.3: стандартные номера IP-портов

Протоколы транспортного уровня идентифицируют трафик вышестоящего уровня с помощью 16-битовых полей, которые называются *номерами портов (port numbers)*. В соединении между двумя устройствами используются порты отправителя и получателя, которые содержатся внутри блока данных протокола. Формат заголовка *протокола пользовательских дейтаграмм (User Datagram Protocol — UDP)* приведен на рис. Б.4, поля с номерами портов отправителя и получателя затенены. Контрольная сумма UDP не является обязательной для протокола IP версии 4. Формат заголовка *протокола управления передачей (Transmission Control Protocol — TCP)* приведен на рис. Б.5, поля с номерами портов отправителя и получателя затенены.

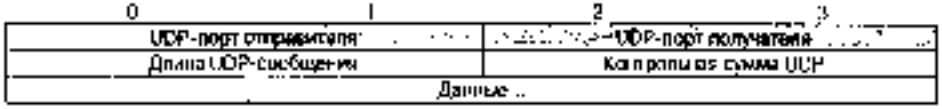


Рис. Б.4. Формат дейтаграммы UDP с полями для номеров портов

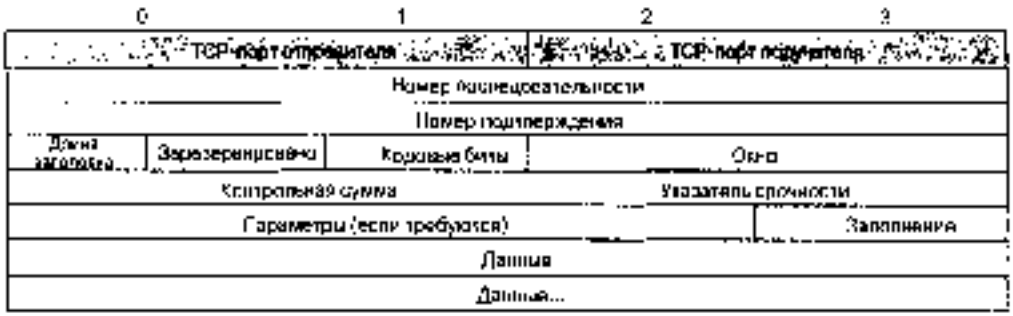


Рис. Б.5. Формат TCP-сегмента с полями для номеров портов

Как UDP-, так и TCP-порты подразделяются на несколько диапазонов:

- стандартные (well-known) номера портов (от 0 до 1023);
- зарегистрированные (registered) номера портов (от 1024 до 49151);
- динамические (dynamic) или частные (private) номера портов (от 49152 до 65535).

Обычно при назначении портов используются общие номера для протоколов UDP и TCP. В соединении от клиента к серверу на стороне последнего используется стандартный порт — *порт запроса службы (service contact port)*, тогда как клиент может свободно динамически назначать собственный номер порта. В протоколе TCP соединение идентифицируется по IP-адресам отправителя и получателя, а также по номерам TCP-портов отправителя и получателя.

Стандартные, или выделенные IP-протоколы, регистрируются в IANA. Информация, представленная здесь, воспроизведена с разрешения организации. Для получения наиболее актуальных сведений о назначении номеров IP-протоколов рекомендуется обратиться к Web-странице www.iana.org/numbers.html, ссылка "Port Numbers" (номера портов).

В табл. Б.3 приведены некоторые широко используемые протоколы, номера их портов и краткое описание. В IANA зарегистрировано около 3350 уникальных номе-

ров портов. Ввиду ограниченного пространства в книге представлен только небольшой набор портов.

Таблица Б.3. Широко используемые протоколы и связанные с ними номера портов

Ключевое слово	Описание	UDP/TCP-порт
echo	Эхо (Echo)	7
discard	Удаление (Discard)	9
systat	Активные пользователи (Active Users)	11
daytime	Время (Daytime, RFC 867)	13
qotd	Цитата дня (Quote of the Day)	17
chargen	Генератор символов (Character Generator)	19
ftp-data	Протокол передачи файлов (стандартные данные)	20
ftp	Протокол передачи файлов (управление)	21
ssh	Протокол удаленной SSH-регистрации (SSH Remote Login Protocol)	22
telnet	Протокол Telnet	23
Любая частная почтовая система	Любая частная почтовая система	24
smtp	Простой протокол передачи почты (Simple Mail Transfer)	25
msg-icp	MSG ICP	29
msg-auth	MSG-аутентификация	31
Любой частный сервер печати	Любой частный сервер печати	35
time	Время (Time)	37
name	Сервер имен узлов (Host Name Server)	42
nameserver	Сервер имен узлов (Host Name Server)	42
nicname	Служба Who Is	43
tacacs	Протокол регистрации узла (Login Host Protocol (TACACS))	49
re-mail-ck	Протокол удаленной проверки почты (Remote Mail Checking Protocol)	50
domain	Сервер доменных имен (Domain Name Server)	53
Любой частный терминальный адрес	Любой частный терминальный адрес	57
Любая частная файловая служба	Любая частная файловая служба	59
whois++	Служба whois++	63
tacacs-ds	Служба базы данных TACACS (TACACS Database Service)	65
sql*net	База Oracle SQL*NET	66
bootps	Сервер начальной загрузки (Bootstrap Protocol Server)	67

Продолжение табл. Б.3

Ключевое слово	Описание	UDP/TCP-порт
bootpc	Клиент начальной загрузки (Bootstrap Protocol Client)	68
tftp	Простой протокол передачи файлов (Trivial File Transfer)	69
gopher	Служба Gopher	70
Любая частная служба набора номера	Любая частная служба набора номера	75
Любая частная RJE-служба	Любая частная RJE-служба	77
finger	Служба Finger	79
http	Протокол HTTP	80
www	Протокол HTTP	80
www-http	Протокол HTTP	80
hosts2-ns	Сервер имен HOSTS2 (HOSTS2 Name Server)	81
xfer	Утилита XFER	82
Любой частный терминальный канал	Любой частный терминальный канал	87
kerberos	Протокол Kerberos	88
dnsix	Таблица маркеров безопасных атрибутов DNSIX (DNSIX Securit Attribute Token Map)	90
npp	Протокол сетевой печати (Network Printing Protocol)	92
dcp	Протокол управления устройствами (Device Control Protocol)	93
objcall	Диспетчер объектов Tivoli (Tivoli Object Dispatcher)	94
acr-nema	Служба ACR-NEMA Digital Imag. & Comm. 300	104
rtelnet	Удаленная Telnet-служба (Remote Telnet Service)	107
snagas	Сервер доступа SNA-шлюза (SNA Gateway Access Server)	108
pop2	Почтовый протокол (Post Office Protocol) версии 2	109
pop3	Почтовый протокол (Post Office Protocol) версии 3	110
sunrpc	Удаленный вызов процедур корпорации SUN (SUN Remote Procedure Call)	111
mcidas	Протокол передачи McIDAS-данных (McIDAS Data Transmission Protocol)	112
ident/auth	Служба аутентификации (Authentication Service)	113
audionews	Многоадресатная передача аудионовостей (Audio News Multicast)	114
sftp	Простой протокол передачи файлов (Simple File Transfer Protocol)	115
uucp-path	Служба UUCP-маршрута (UUCP Path Service)	117
sqlserv	SQL-службы	118

Продолжение табл. Б.3

Ключевое слово	Описание	UDP/TCP-порт
nntp	Протокол передачи сетевых новостей (Network News Transfer Protocol)	119
ntp	Синхронизирующий сетевой протокол (Network Time Protocol)	123
pwdgen	Протокол генератора паролей (Password Generator Protocol)	129
cisco-fna	Протокол Cisco FNATIVE	130
cisco-tna	Протокол Cisco TNATIVE	131
cisco-sys	Протокол Cisco SYSMAINT	132
ingres-net	Служба INGRES-NET	134
profile	Служба именования PROFILE	136
netbios-ns	Служба имен NetBIOS (NetBIOS Name Service)	137
netbios-dgm	Служба дейтаграмм NetBIOS (NetBIOS Datagram Service)	138
netbios-ssn	Сеансовая служба NetBIOS (NetBIOS Session Service)	139
imap	Протокол доступа к сообщениям в сети Internet (Internet Message Access Protocol)	143
sql-net	Протокол SQL-NET	150
sgmp	Протокол SGMP	153
sqlsrv	Служба SQL	156
pcmail-srv	Сервер PCMail	158
sgmp-traps	Протокол SGMP-TRAPS	160
snmp	Протокол SNMP	161
snmptrap	Протокол SNMPTRAP	162
cmip-man	Диспетчер CMIP/TCP	163
send	SEND	169
print-srv	Протокол Network PostScript	170
xyplex-mux	Протокол Xyplex	173
mailq	Протокол MAILQ	174
vmnet	Протокол VMNET	175
xdmcp	Управляющий протокол менеджера экрана в системе X (X Display Manager Control Protocol)	177
bgp	Граничный шлюзовый протокол (Border Gateway Protocol)	179
mumps	Протокол Plus Five's MUMPS	188
irc	Протокол разговора по Internet (Internet Relay Chat Protocol)	194
dnsix	Аудит модуля сетевого уровня DNSIX (DNSIX Network Level Module Audit)	195

Продолжение табл. Б.3

Ключевое слово	Описание	UDP/TCP-порт
dnsix-smm-red	Перенаправление сеанса аудита модуля сетевого уровня DNSIX (DNSIX Session Management Module Audit Redirect)	196
dls	Служба расположения каталогов (Directory Location Service)	197
dls-mon	Монитор службы расположения каталогов (Directory Location Service Monitor)	198
src	Контроллер системных ресурсов IBM (IBM System Resource Controller)	200
at-rtmp	Поддержка маршрутизации AppleTalk (AppleTalk Routing Maintenance)	201
at-nbp	Привязка имен AppleTalk (AppleTalk Name Binding)	202
at-3	AppleTalk не используется	203
at-echo	AppleTalk-эхо	204
at-5	AppleTalk не используется	205
at-zis	Зонная информация AppleTalk (AppleTalk Zone Information)	206
at-7	AppleTalk не используется	207
at-8	AppleTalk не используется	208
qmtp	Протокол быстрой передачи почты (The Quick Mail Transfer Protocol)	209
ipx	Протокол IPX	213
vmpwscs	Протокол VM PWSCS	214
softpc	Протокол Insignia Solutions	215
dbase	Протокол dBASE UNIX	217
imap3	Протокол интерактивного доступа к почте (Interactive Mail Access Protocol) версия 3	220
http-mgmt	Протокол http mgmt	280
asip-webadmin	Протокол AppleShare IP WebAdmin	311
pip-event	Протокол PTP Event	319
ptp-general	Протокол PTP General	320
pdap	Протокол доступа к данным Prospero (Prospero Data Access Protocol)	344
rsvp_tunnel	RSVP-туннель	363
rpc2portmap	Протокол rpc2portmap	369
aurp	Протокол маршрутизации AppleTalk на основе обновлений (AppleTalk Update-Based Routing Protocol)	387
ldap	Упрощенный протокол доступа к каталогам (Lightweight Directory Access Protocol)	389
netcp	Протокол управления NETscout (NETscout Control Protocol)	395

Ключевое слово	Описание	UDP/TCP-порт
netware-ip	Сеть Novell NetWare в IP	396
ups	Источник бесперебойного питания (Uninterruptible power supply)	401
smsp	Протокол служб управления хранилищем данных (Storage Management Services Protocol)	413
mobileip-agent	MobileIP-агент	434
mobilip-mn	MobilIP MN	435
https	Протокол HTTP поверх TLS/SSL	443
snpp	Протокол SNPP	444
microsoft-ds	Протокол Microsoft-DS	445
appleqtc	Протокол Apple QuickTime	458
ss7ns	ss7ns	477
p1	Служба P1	481
isakmp	Протокол isakmp	500
exec	Удаленное выполнение процесса (Remote process execution)	512
login	Удаленная регистрация с помощью Telnet (remote login by Telnet)	513
shell	Протокол cmd	514
printer	Спул принтера	515
ntalk	Протокол ntalk	518
utime	Протокол unixtime	519
ncp	Протокол NCP	524
timed	Протокол сервера синхронизации времени	525
irc-serv	Протокол IRC-SERV	529
courier	Протокол rpc	530
conference	Протокол chat	531
netnews	Протокол readnews	532
netwall	Для широковещания экстренных служб (For emergency broadcasts)	533
iiop	Протокол iiop	535
nntsp	Потоковый протокол сетевой среды (Networked Media Streaming Protocol)	537
uucp	Протокол uucpd	540
uucp-rlogin	Протокол uucp rlogin	541
klogin	Протокол klogin	543
kshell	Протокол kromd	544
appleqtcsrvr	Протокол Appleqtcsrvr	545
dhcpv6-client	Клиент DHCP версии 6	546

Продолжение табл. Б.3

Ключевое слово	Описание	UDP/TCP-порт
dhcpv6-server	Сервер DHCP версии 6	547
afpovertcp	Протокол AFC поверх TCP	548
rtsp	Протокол управления потоком реального времени (Real Time Stream Control Protocol)	554
remotefs	rfs-сервер	556
rmonitor	Служба удаленного мониторинга	560
monitor	Мониторинг	561
nntps	Протокол NNTP поверх TLS/SSL (ранее snntp)	563
whoami	Протокол whoami	565
snpp-heartbeat	Протокол SNPP HEARTBEAT	580
imap4-ssl	IMAP4 + SSL (рекомендуется использовать 993)	585
password-chg	Изменение пароля	586
eudora-set	Протокол Eudora Set	592
http-rpc-epmap	Протокол HTTP RPC Ep Map	593
sco-websrvrmg3	Протокол управления Web-сервером SCO версии 3	598
ipcserver	IPC-сервер SUN	600
sshell	Протокол SSLshell	614
sco-inetmgr	Диспетчер настройки Internet (Internet Configuration Manager)	615
sco-sysmgr	Сервер системного администрирования SCO (SCO System Administration Server)	616
sco-dtmgr	Сервер администрирования настольных систем SCO (SCO Desktop Administration Server)	617
sco-websrvrmgr	Протокол управления Web-сервером SCO	620
ldaps	Протокол LDAP поверх TLS/SSL (ранее sldap)	636
dhcp-failover	Восстановление после сбоев DHCP	647
mac-srvr-admin	Протокол администрирования сервера MacOS	660
doom	Игра Doom компании Id Software	666
corba-iiop	Протокол CORBA IIOP	683
corba-iiop-ssl	Протокол CORBA IIOP SSL	684
nmap	Протокол NMAP	689
msexch-routing	Маршрутизация MS Exchange	691
ieee-mms-ssl	Протокол IEEE-MMS-SSL	695
cisco-tdp	Протокол Cisco TDP	711
edm	Гибкий диспетчер лицензий (Flexible License Manager)	744
kerberos-adm	Администрирование Kerberos	749
phonebook	Протокол Phone	767
dhcp-failover2	Протокол восстановления DHCP-сервера версии 2	847

Продолжение табл. Б.3

Ключевое слово	Описание	UDP/TCP-порт
ftps-data	Протокол FTP, данные поверх TLS/SSL	989
ftps	Протокол FTP, управление поверх TLS/SSL	990
nas	Система администрирования Netnews (Netnews Administration System)	991
telnets	Протокол Telnet поверх TLS/SSL	992
imaps	Протокол IMAP4 поверх TLS/SSL	993
ircs	Протокол IRC поверх TLS/SSL	994
pop3s	Протокол POP3 поверх TLS/SSL (ранее spop3)	995
sunclustermgr	Управление кластерами корпорации SUN (Cluster Manager)	1097
tripwire	Протокол TRIPWIRE	1169
shockwave2	Протокол Shockwave 2	1257
h323hostcallsc	Безопасный вызов узла H323 (H323 Host Call Secure)	1300
lotusnote	Протокол Lotus Notes	1352
novell-lu6.2	Протокол Novell LU6.2	1416
ms-sql-s	Сервер баз данных Microsoft SQL	1433
ms-sql-m	Монитор сервера Microsoft SQL	1434
ibm-cics	Протокол IBM CICS	1435
sybase-sqlany	Протокол Sybase SQL Any	1498
shivadiscovery	Протокол Shiva	1502
wins	Служба Internet-имен Microsoft Windows (Microsoft Windows Internet Name Service)	1512
ingreslock	Протокол Ingres	1524
orasrv	Протокол корпорации Oracle	1525
tlisrv	Протокол корпорации Oracle	1527
coauthor	Протокол корпорации Oracle	1529
rdb-dbs-disp	Удаленная база данных Oracle (Oracle Remote Data Base)	1571
oraclenames	Протокол имен корпорации Oracle	1575
oninime	Протокол oninit	1622
shockwave	Протокол Shockwave	1626
oraclenet8cman	Протокол Oracle Net8 Cman	1630
cert-initiator	Протокол инициации сертификата	1639
cert-responder	Протокол ответа на сертификат	1640
kermit	Протокол kermit	1649
groupwise	Протокол groupwise	1677
rsvp-encap-1	Протокол RSVP-ENCAPSULATION-1	1698
rsvp-encap-2	Протокол RSVP-ENCAPSULATION-2	1699
h323gatedisc	Протокол обнаружения шлюзов H323	1718

Ключевое слово	Описание	UDP/TCP-порт
h323gatestat	Протокол статистики шлюза H323	1719
h323hostcall	Протокол вызова узла H323	1720
cisco-net-mgmt	Протокол cisco-net-mgmt	1741
oracle-em1	Протокол oracle-em1	1748
oracle-em2	Протокол oracle-em2	1754
ifip-mcast	Протокол многоадресатного ifip-сервера	1758
www-ldap-gw	Протокол шлюза www-ldap	1760
bmc-net-admin	Протокол bmc-net-admin	1769
bmc-net-svc	Протокол bmc-net-svc	1770
oracle-vp2	Протокол Oracle-VP2	1808
oracle-vp1	Протокол Oracle-VP1	1809
radius	Протокол RADIUS	1812
radius-acct	Учет службы RADIUS	1813
hsrp	Протокол резервного маршрутизатора (Hot Standby Router Protocol)	1985
licensedaemon	Управление лицензиями Cisco (Cisco license management)	1986
tr-rsrb-p1	Порт Cisco RSRP Priority 1	1987
tr-rsrb-p2	Порт Cisco RSRP Priority 2	1988
tr-rsrb-p3	Порт Cisco RSRP Priority 3	1989
stun-p1	Порт Cisco STUN Priority 1	1990
stun-p2	Порт Cisco STUN Priority 2	1991
stun-p3	Порт Cisco STUN Priority 3	1992
snmp-tcp-port	TCP-порт протокола Cisco SNMP	1993
stun-port	Порт последовательного туннеля Cisco	1994
perf-port	perf-порт Cisco	1995
tr-rsrb-port	Порт удаленной SRB-маршрутизации Cisco	1996
gdp-port	Протокол обнаружения шлюзов Cisco (Cisco Gateway Discovery Protocol)	1997
x25-svc-port	Служба Cisco X.25 (XOT)	1998
tcp-id-port	Порт идентификации устройств Cisco (Cisco identification port)	1999
dlsrpn	Номер порта считывания коммутатора канального уровня (Data Link Switch Read Port Number)	2065
dlswpn	Номер порта записи коммутатора канального уровня (Data Link Switch Write Port Number)	2067
ah-esp-encap	Протоколы AH и ESP, инкапсулированные в UDP-пакет	2070
h2250-annex-g	Протокол H.225.0 Annex G	2099
ms-olap3	Протокол Microsoft OLAP	2382

Ключевое слово	Описание	UDP/TCP-порт
ovsessionmgr	Диспетчер сеансов OpenView (OpenView Session Manager)	2389
ms-olap1	Протокол MS OLAP 1	2393
ms-olap2	Протокол MS OLAP 2	2394
mgcp-gateway	Шлюз протокола управления шлюза-носителя (Media Gateway Control Protocol Gateway)	2427
ovwdb	Служба OpenView NNM	2447
giop	Протокол Oracle GIOP	2481
giop-ssl	Протокол Oracle GIOP SSL	2482
ttc	Протокол Oracle TTC	2483
ttc-ssl	Протокол Oracle TTC SSL	2484
citrixima	Протокол Citrix IMA	2512
citrixadmin	Протокол Citrix ADMIN	2513
call-sig-trans	Сигнальный транспорт вызовов H.323 Annex E	2517
windb	Протокол WinDb	2522
novell-zen	Протокол Novell ZEN	2544
clp	Протокол линий Cisco (Cisco Line Protocol)	2567
hl7	Протокол HL7	2575
citrixmaclient	Клиент Citrix MA	2598
sybasesanywhere	Протокол Sybase Anywhere	2638
novell-ipx-cmd	Протокол Novell IPX CMD	2645
sms-rcinfo	Протокол SMS RCINFO	2701
sms-xfer	Протокол SMS XFER	2702
sms-chat	Протокол SMS CHAT	2703
sms-remctrl	Протокол SMS REMCTRL	2704
mgcp-callagent	Агент вызова протокола управления шлюза-носителя (Media Gateway Control Protocol Call Agent)	2727
dicom-iscl	Протокол DICOM ISCL	2761
dicom-tls	Протокол DICOM TLS	2762
citrix-rtmp	Протокол Citrix RTMP	2897
wap-push	Протокол WAP Push	2948
wap-pushsecure	Протокол WAP Push Secure	2949
h263-video	Потоковое видео H.263	2979
lotusmtap	Протокол агента отслеживания почты Lotus (Lotus Mail Tracking Agent Protocol)	3007
nfsd	Синхронные службы NetWare	3092
bmcpatrolagent	Агент BMC Patrol	3181
bmcpatrolrnvu	Точка встречи BMC Patrol	3182
ccmail	Протокол cc:mail/lotus	3264

Ключевое слово	Описание	UDP/TCP-порт
msft-gc	Протокол глобального каталога корпорации Microsoft (Microsoft Global Catalog)	3268
msft-gc-ssl	Протокол глобального каталога корпорации Microsoft с поддержкой LDAP/SSL	3269
Неавторизованное использование SAP R/3	Неавторизованное использование SAP R/3	с 3300 по 3301
mysql	Протокол MySQL	3306
ms-cluster-net	Протокол MS Cluster Net	3343
ssql	Протокол SSQL	3352
ms-wbt-server	Протокол сервера MS WBT	3389
mira	Протокол удаленного доступа Apple (Apple Remote Access Protocol)	3454
prsvp	Порт RSVP	3455
patrolview	Протокол Patrol View	4097
vrml-multi-use	Многопользовательские системы VRML	с 4200 по 4299
rwhois	Удаленная служба Who Is	4321
bmc-reporting	Служба отчетов BMC	4568
sip	Протокол SIP	5060
sip-tls	Протокол SIP-TLS	5061
pcanywheredata	Протокол PcANYWHEREdata	5631
pcanywherestat	Протокол pcANYWHEREstat	5632
x11	Система X Window	с 6000 по 6063
bmc-grx	Протокол BMC GRX	6300
bmc-perf-agent	Агент BMC PERFORM AGENT	6767
bmc-perf-mgrd	Служба BMC PERFORM MGRD	6768
sun-lm	Диспетчер лицензий корпорации SUN (SUN License Manager)	7588
http-alt	Альтернативный порт протокола HTTP (см. порт 80)	8080
cp-cluster	Протокол контрольных точек (Check Point) кластеров	8116
patrol	Протокол Patrol	8160
patrol-snmp	Протокол Patrol SNMP	8161
wap-wsp	Независящая от соединения сеансовая WAP-служба	9200
wap-wsp-wtp	Сеансовая WAP-служба	9201
wap-wsp-s	Безопасная независящая от соединения сеансовая WAP-служба	9202
wap-wsp-wtp-s	Безопасная сеансовая WAP-служба	9203
wap-vcard	Протокол WAP vCard	9204
wap-vcal	Протокол WAP vCal	9205
wap-vcard-s	Безопасный протокол WAP vCard	9206

Окончание табл. Б.3

Ключевое слово	Описание	UDP/TCP-порт
wap-vcal-s	Безопасный протокол WAP vCal	9207
bmc-perf-sd	Служба BMC-PERFORM-SERVICE DAEMON	10128
h323callsigalt	Замена сигнализации H323-вызова	11720
vofr-gateway	VoFR-шлюз	21590
quake	Игра Quake	26000
flex-lm	Протокол FLEX LM (1–10)	с 27000 по 27009
traceroute	Используется утилитой traceroute	33434
reachout	Протокол REACHOUT	43188

Б.4: стандартные IP-адреса многоадресатного вещания

В некоторых клиент-серверных приложениях для того чтобы отправлять большие потоки данных многим узлам в ходе одной передачи, используется многоадресатный пакет (multicast packet). В многоадресатном пакете для обмена данными с клиентами, сконфигурированными для получения таких пакетов, используется специальная адресация второго и третьего уровней. Многоадресатный пакет содержит IP-адреса класса D для указания группы устройств, которые должны получить этот пакет. Такая группа называется *многоадресатной (multicast group)*, а IP-адрес транслируется непосредственно в многоадресатный Ethernet-адрес. Первые 24 бита многоадресатного Ethernet-адреса равны `01-00-5E`, следующий бит равен нулю, а последние 23 бита устанавливаются таким образом, что совпадают с 23 младшими битами многоадресатного IP-адреса. На рис. Б.6 иллюстрируется преобразование многоадресатных адресов третьего уровня в Ethernet-адреса второго уровня.

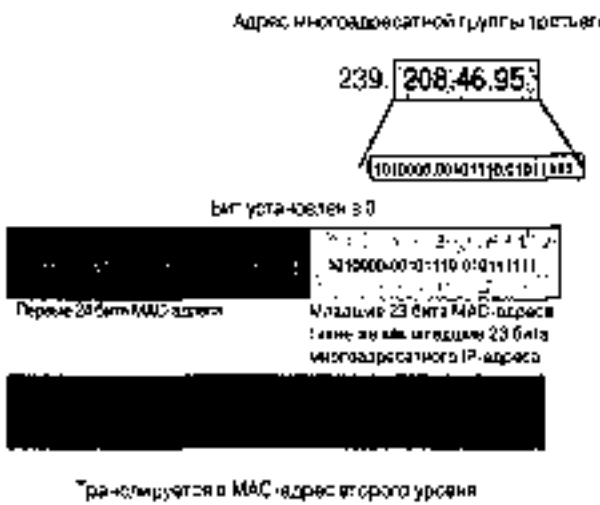


Рис. Б.6. Многоадресатное преобразование адресов второго и третьего уровней

Стандартные или выделенные IP-протоколы регистрируются в организации IANA. Информация, представленная здесь, воспроизведена с разрешения организации. Для получения наиболее актуальных сведений о назначении номеров IP-протоколам рекомендуется обратиться к Web-странице www.iana.org/numbers.html, ссылка "Multicast Addresses".

Расширения узла для многоадресатного IP-вещания (RFC 1112) определяют требуемые расширения реализации Internet-протокола в узле для поддержки многоадресатной технологии. Многоадресатные адреса находятся в диапазоне 224.0.0.0–239.255.255.255. Текущие адреса перечислены в таблице ниже.

Диапазон адресов 224.0.0.0—224.0.0.255 включительно зарезервирован для использования протоколов маршрутизации и других протоколов низкоуровневого изучения топологии или поддержки, таких, как протокол обнаружения шлюзов и протокол отчетов о составе групп. Многоадресатные маршрутизаторы не должны осуществлять пересылку многоадресатной дейтаграммы на адреса отправителей, находящиеся в этом диапазоне, независимо от величины TTL данной дейтаграммы.

В табл. Б.4 приведены зарегистрированные многоадресатные адреса наряду с приложениями и номерами документов RFC (если они применимы) или другой справочной информацией.

Таблица Б.4. Зарегистрированные многоадресатные адреса, а также связанные с ними приложения, документы RFC и справочная информация

Группа, приложение и ссылки	Адрес
Базовый адрес (зарезервирован) RFC 1112	224.0.0.0
Все системы в данной подсети RFC 1112	224.0.0.1
Все маршрутизаторы в данной подсети	224.0.0.2
Не выделен	224.0.0.3
Протокол DVMRP Маршрутизаторы RFC 1075	224.0.0.4
Протокол OSPFIGP Протокол OSPFIGP — все маршрутизаторы RFC 2328	224.0.0.5
Протокол OSPFIGP Протокол OSPFIGP — выделенные маршрутизаторы RFC 2328	224.0.0.6
Протокол ST Маршрутизаторы RFC 1190	224.0.0.7
Протокол ST Узлы RFC 1190	224.0.0.8
Протокол RIP2 Маршрутизаторы RFC 1723	224.0.0.9

Продолжение табл. Б.4

Группа, приложение и ссылки	Адрес
Протокол EIGRP Маршрутизаторы	224.0.0.10
Мобильные агенты	224.0.0.11
Протокол DHCP Агент сервер/ретранслятор RFC 1884	224.0.0.12
Все PIM-маршрутизаторы	224.0.0.13
Протокол RSVP-ENCAPSULATION	224.0.0.14
Все cbt-маршрутизаторы	224.0.0.15
Выделенный экземпляр sbm	224.0.0.16
Все sbm	224.0.0.17
Протокол VRRP	224.0.0.18
IPAllL1Iss	224.0.0.19
IPAllL2Iss	224.0.0.20
IPAllIntermediate Systems	224.0.0.21
Протокол IGMP	224.0.0.22
GLOBECAST-ID	224.0.0.23
Невыделенный	224.0.0.24
Маршрутизатор-коммутатор	224.0.0.25
Невыделенный	224.0.0.26
Протокол AI MPP Hello	224.0.0.27
ETC-управление	224.0.0.28
GE-FANUC	224.0.0.29
indigo-vhdp	224.0.0.30
Протокол Shinbroadband	224.0.0.31
Протокол Digistar	224.0.0.32
ll-system-management	224.0.0.33
Протокол pt2-discover	224.0.0.34
Протокол DXCLUSTER	224.0.0.35
DTCP-анонс	224.0.0.36
Zeroconfaddr	с 224.0.0.37 по 224.0.0.68
Зарезервированы	с 224.0.0.69 по 224.0.0.100
cisco-mhap	224.0.0.101
Протокол HSRP	224.0.0.102
Протокол MDAP	224.0.0.103
Невыделенные	с 224.0.0.104 по 224.0.0.250

Продолжение табл. Б.4

Группа, приложение и ссылки	Адрес
mDNS	224.0.0.251
Невыделенные	с 224.0.0.25 по 224.0.0.255
Группа VMTP-диспетчеров RFC 1045	224.0.1.0
NTP (Network Time Protocol — синхронизирующий сетевой протокол) RFC 1119	224.0.1.1
SGI-Dog ght	224.0.1.2
Rwhod	224.0.1.3
Протокол VNP	224.0.1.4
Artificial Horizons — Aviator	224.0.1.5
Сервер NSS (Name Service Server — сервер службы имен)	224.0.1.6
Рассылка AUDIONEWS (Audio News Multicast — многоадресатная доставка аудионовостей)	224.0.1.7
Информационная служба SUN NIS+	224.0.1.8
Протокол MTP (Multicast Transport Protocol — протокол многоадресатной транспортировки)	224.0.1.9
IETF-1-LOW-AUDIO	224.0.1.10
IETF-1-AUDIO	224.0.1.11
IETF-1-VIDEO	224.0.1.12
IETF-2-LOW-AUDIO	224.0.1.13
IETF-2-AUDIO	224.0.1.14
IETF-2-VIDEO	224.0.1.15
MUSIC-SERVICE	224.0.1.16
SEANET-TELEMETRY	224.0.1.17
SEANET-IMAGE	224.0.1.18
MLOADD	224.0.1.19
Любой частный эксперимент	224.0.1.20
Протокол DVMRP поверх MOSPF	224.0.1.21
SVRLOC	224.0.1.22
XINGTV	224.0.1.23
Протокол microsoft-ds	224.0.1.24
nbc-pro	224.0.1.25
nbc-pln	224.0.1.26
Imsc-calren-1	224.0.1.27
Imsc-calren-2	224.0.1.28
Imsc-calren-3	224.0.1.29
Imsc-calren-4	224.0.1.30
amgr-mfo	224.0.1.31

Продолжение табл. Б.4

Группа, приложение и ссылки	Адрес
Протокол Mtrace	224.0.1.32
Протокол RSVP-encap-1	224.0.1.33
Протокол RSVP-encap-2	224.0.1.34
SVRLOC-DA	224.0.1.35
rln-сервер	224.0.1.36
proshare-mc	224.0.1.37
Dantz	224.0.1.38
cisco-rp-анонс	224.0.1.39
cisco-rp-обнаружение	224.0.1.40
gatekeeper	224.0.1.41
iberiagames	224.0.1.42
nwn-обнаружение	224.0.1.43
nwn-адаптер	224.0.1.44
isma-1	224.0.1.45
isma-2	224.0.1.46
Telerate	224.0.1.47
Ciena	224.0.1.48
dcap-серверы RFC 2114	224.0.1.49
dcap-клиенты RFC 2114	224.0.1.50
mcntp-каталог	224.0.1.51
mbone-vcr-каталог	224.0.1.52
Тестовые сообщения (Heartbeat)	224.0.1.53
sun-mc-grp	224.0.1.54
extended-sys	224.0.1.55
Pericos	224.0.1.56
ins-adv-multi	224.0.1.57
vcals-dmu	224.0.1.58
Zube	224.0.1.59
hp-device-disc	224.0.1.60
tms-production	224.0.1.61
Sunscalar	224.0.1.62
mmtp-cnpoc	224.0.1.63
compaq-peer	224.0.1.64
iapp	224.0.1.65
multihasc-com	224.0.1.66
serv-обнаружение	224.0.1.67

Продолжение табл. Б.4

Группа, приложение и ссылки	Адрес
Протокол Mdhcpdiscover RFC 2730	224.0.1.68
Протокол MMP-bundle-discovery1	224.0.1.69
Протокол MMP-bundle-discovery2	224.0.1.70
Передача данных XYPOINT DGPS	224.0.1.71
GilatSkySurfer	224.0.1.72
SharesLive	224.0.1.73
NorthernData	224.0.1.74
Протокол SIP	224.0.1.75
IAPP	224.0.1.76
AGENTVIEW	224.0.1.77
Протокол Tibco Multicast1	224.0.1.78
Протокол Tibco Multicast2	224.0.1.79
Протокол MSP	224.0.1.80
OTT (One-way Trip Time — время односторонней передачи)	224.0.1.81
TRACKTICKER	224.0.1.82
ctn-mc	224.0.1.83
jini-анонс	224.0.1.84
jini-запрос	224.0.1.85
sde-обнаружение	224.0.1.86
DirecPC-SI	224.0.1.87
B1Rmonitor	224.0.1.88
3Com-AMP3 dRMON	224.0.1.89
imFtmSvc	224.0.1.90
NQDS4	224.0.1.91
NQDS5	224.0.1.92
NQDS6	224.0.1.93
NLVL12	224.0.1.94
NTDS1	224.0.1.95
NTDS2	224.0.1.96
NODSA	224.0.1.97
NODSB	224.0.1.98
NODSC	224.0.1.99
NODSD	224.0.1.100
NQDS4R	224.0.1.101
NQDS5R	224.0.1.102
NQDS6R	224.0.1.103
NLVL12R	224.0.1.104

Продолжение табл. Б.4

Группа, приложение и ссылки	Адрес
NQDS1R	224.0.1.105
NQDS2R	224.0.1.106
NQDSAR	224.0.1.107
NQDSBR	224.0.1.108
NQDSCR	224.0.1.109
NQDSDR	224.0.1.110
MRM	224.0.1.111
TVE-FILE	224.0.1.112
TVE-ANNOUNCE	224.0.1.113
Mac Srv Loc	224.0.1.114
Простое многоадресатное вещание (Simple Multicast)	224.0.1.115
SpectraLinkGW	224.0.1.116
Dialbroadcast	224.0.1.117
Tivoli Systems	224.0.1.118
pq-lic-mcast	224.0.1.119
HYPERFEED	224.0.1.120
Pipesplatform	224.0.1.121
LiebDevMgmg-DM	224.0.1.122
TRIBALVOICE	224.0.1.123
Невыделенный (использование прекращено 1/29/01)	224.0.1.124
PolyCom Relay1	224.0.1.125
Infront Multi1	224.0.1.126
XINX DEVICE DISC	224.0.1.127
CNN	224.0.1.128
Основной Протокол PTP	224.0.1.129
Замена протокола PTP 1	224.0.1.130
Замена протокола PTP 2	224.0.1.131
Замена протокола PTP 3	224.0.1.132
Протокол ProCast	224.0.1.133
3Com Disco	224.0.1.134
CS-Multicasting	224.0.1.135
TS-MC-1	224.0.1.136
Make Source	224.0.1.137
Telebolsa	224.0.1.138
SUMACong	224.0.1.139
Невыделенный	224.0.1.140
Сервера DHCP	224.0.1.141
CN Router-LL	224.0.1.142

Группа, приложение и ссылки	Адрес
EMWIN	224.0.1.143
Alchemy Cluster	224.0.1.144
Salcast One	224.0.1.145
Salcast Two	224.0.1.146
Salcast Three	224.0.1.147
Inline	224.0.1.148
8x8 Multicast	224.0.1.149
Невыделенный	224.0.1.150
Intline-1	224.0.1.151
Intline-2	224.0.1.152
Intline-3	224.0.1.153
Intline-4	224.0.1.154
Intline-5	224.0.1.155
Intline-6	224.0.1.156
Intline-7	224.0.1.157
Intline-8	224.0.1.158
Intline-9	224.0.1.159
Intline-10	224.0.1.160
Intline-11	224.0.1.161
Intline-12	224.0.1.162
Intline-13	224.0.1.163
Intline-14	224.0.1.164
Intline-15	224.0.1.165
marratech-ec	224.0.1.166
EMS-InterDev	224.0.1.167
ltb301	224.0.1.168
rtv-аудио	224.0.1.169
rtv-видео	224.0.1.170
HAVI-Sim	224.0.1.171
Nokia Cluster	224.0.1.172
host-аллрос	224.0.1.173
host-анонс	224.0.1.174
plk-кластер	224.0.1.175
Протокол Proxim	224.0.1.176
Невыделенные	с 224.0.1.177 по 224.0.0.255
Группа "who" системы BSD (неофициальный)	224.0.2.1
SUN RPC PMAPPROC_CALLIT	224.0.2.2
Служба SIAC MDD	с 224.0.2.64 по 224.0.2.95

Продолжение табл. Б.4

Группа, приложение и ссылки	Адрес
CoolCast	с 224.0.2.96 по 224.0.2.127
WOZ-Garage	с 224.0.2.128 по 224.0.2.191
SIAC MDD Market Service	с 224.0.2.192 по 224.0.2.255
Общая служба RFE	с 224.0.3.0 по 224.0.3.255
Индивидуальные конференции RFE	с 224.0.4.0 по 224.0.4.256
CDPD-группы	с 224.0.5.0 по 224.0.5.127
SIAC Market Service	с 224.0.5.128 по 224.0.5.191
Невыделенные IANA	с 224.0.5.192 по 224.0.5.255
Cornell ISIS Project	с 224.0.6.0 по 224.0.6.127
Невыделенные IANA	с 224.0.6.128 по 224.0.6.256
Адресные запросы (Where-Are-You)	с 224.0.7.0 по 224.0.7.256
INTV	с 224.0.8.0 по 224.0.8.255
Invisible Worlds	с 224.0.9.0 по 224.0.9.255
DLSw-группы	с 224.0.10.0 по 224.0.10.255
NCC.NET-аудио	с 224.0.11.0 по 224.0.11.256
Microsoft и MSNBC	с 224.0.12.0 по 224.0.12.63
Сетевые новости UUNET PIPEX	с 224.0.13.0 по 223.0.13.256
NLANR	с 224.0.14.0 по 224.0.14.255
Hewlett Packard	с 224.0.15.0 по 224.0.15.255
XingNet	с 224.0.16.0 по 224.0.16.255
Mercantile & Commodity Exchange	с 224.0.17.0 по 224.0.17.31
NDQMD1	с 224.0.17.32 по 224.0.17.63
QDN-DTV	с 224.0.17.64 по 224.0.17.127
Dow Jones	с 224.0.18.0 по 224.0.18.255
Walt Disney Company	с 224.0.19.0 по 224.0.19.63
Cal Multicast	с 224.0.19.64 по 224.0.19.95
SIAC Market Service	с 224.0.19.96 по 224.0.19.127
IIG Multicast	с 224.0.19.128 по 224.0.19.191
Metropol	с 224.0.19.192 по 224.0.19.207
XenoScience, Inc.	с 224.0.19.208 по 224.0.19.239
HYPERFEED	с 224.0.19.240 по 224.0.19.255
MS-IP/TV	с 224.0.20.0 по 224.0.20.63
Reliable Network Solutions	с 224.0.20.64 по 224.0.20.127
Группа TRACKTICKER	с 224.0.20.128 по 224.0.20.143
CNR Rebroadcast MCA	с 224.0.20.144 по 224.0.20.207
Talarian MCAST	с 224.0.21.0 по 224.0.21.127

Окончание табл. Б.4

Группа, приложение и ссылки	Адрес
WORLD MCAST	с 224.0.22.0 по 224.0.22.255
Ограниченная группа домена (Domain Scoped Group)	с 224.0.252.0 по 224.0.252.255
Группа отчетов (Report Group)	с 224.0.253.0 по 224.0.253.255
Группа запросов (Query Group)	с 224.0.254.0 по 224.0.254.255
Граничные маршрутизаторы	с 224.0.255.0 по 224.0.255.255
Многоадресатные группы ST RFC 1190	с 224.1.0.0 по 224.1.255.255
Мультимедийные конференц-вызовы (Multimedia Conference Calls)	с 224.2.0.0 по 224.2.127.253
Анонсы SAPv1	224.2.127.254
Анонсы SAPv0 (не был утверждён)	224.2.127.255
Динамические назначения SAP	с 224.2.128.0 по 224.2.255.255
Временные группы DIS	с 224.252.0.0 по 224.255.255.255
MALLOC (временные – восстановить 1/01)	с 225.0.0.0 по 225.255.255.255
Временные группы VMTP	с 232.0.0.0 по 232.255.255.255
Статическое распределение (временные – восстановить 03/02)	с 233.0.0.0 по 233.255.255.255
Административно ограниченные IANA RFC 2365	с 239.0.0.0 по 239.255.255.255
Зарезервированы IANA	с 239.0.0.0 по 239.63.255.255
Зарезервированы IANA	с 239.64.0.0 по 239.127.255.255
Зарезервированы IANA	с 239.128.0.0 по 239.191.255.255
Локальный диапазон организации RFC 2365	с 239.192.0.0 по 239.251.255.255
Локальный диапазон сегмента (зарезервированы) RFC 2365	с 239.252.0.0 по 239.252.255.255
Локальный диапазон сегмента (зарезервированы) RFC 2365	с 239.253.0.0 по 239.253.255.255
Локальный диапазон сегмента (зарезервированы) RFC 2365	с 239.254.0.0 по 239.254.255.255
Локальный диапазон сегмента RFC 2365	с 239.255.0.0 по 239.255.255.255
rwaldv	239.255.2.2

Б.5: коды Ethernet

Список широко используемых кодов Ethernet поддерживается организацией IANA. Информация, представленная здесь, воспроизведена с разрешения организации. Для

получения наиболее актуальных сведений о назначении кодовых чисел Ethernet рекомендуется обратиться к Web-странице www.iana.org/numbers.htm, ссылка "Ethernet Numbers" (номера Ethernet). В табл. Б.5 приведены кодовые числа Ethernet в шестнадцатеричном формате, а также их описания.

Таблица Б.5. Коды Ethernet

Шестнадцатеричное значение	Описание
с 0000 по 05DC	Поле длины IEEE 802.3
с 0101 по 01FF	Экспериментальные
200	XEROX PUP (см. 0A00)
201	PUP Addr Trans (см. 0A01)
400	Nixdorf
600	XEROX NS IDP
660	DLOG
661	DLOG
800	Internet IP (IPv4)
801	X.75 Internet
802	NBS Internet
803	ECMA Internet
804	Chaosnet
805	X.25 Level 3
806	ARP
807	XNS-совместимость
808	Frame Relay ARP (RFC 1701)
081C	Symbolics Private
с 0888 по 088A	Xyplex
900	Отладка сети Ungermann-Bass
0A00	Xerox IEEE802.3 PUP
0A01	Адресное преобразование PUP
0BAD	Banyan VINES
0BAE	Петля VINES (RFC 1701)
0BAF	VINES-эхо (RFC 1701)
1000	Согласование трейлера Berkeley
с 1001 по 100F	Инкапсуляция IP трейлера Berkeley
1600	Действительные системы
4242	Протокол базового блока PCS (PCS Basic Block Protocol)
5208	BBN Simnet
6000	DEC немыделенный (экспериментальный)
6001	DEC MOP Dump/Load
6002	DEC MOP удаленная консоль

Шестнадцатеричное значение	Описание
6003	DEC DECNET маршрут фаза IV
6004	DEC LAT
6005	Протокол диагностики DEC (DEC Diagnostic Protocol)
6006	Клиентский протокол DEC (DEC Customer Protocol)
6007	DEC LAVC, SCA
с 6008 по 6009	DEC невыделенный
с 6010 по 6014	3Com Corporation
6558	Trans Ether Bridging (RFC 1701)
6559	Бесформатный протокол Frame Relay (Raw Frame Relay, RFC 1701)
7000	Загрузка Ungermann-Bass
7002	Ungermann-Bass dia/loop
с 7020 по 7029	LRT
7030	Proteon
7034	Cabletron
8003	Cronus VLN
8004	Cronus Direct
8005	HP Probe
8006	Nestar
8008	AT&T
8010	Excelan
8013	Диагностика SGI
8014	Сетевые игры SGI
8015	SGI зарезервированный
8016	Сервер отражения SGI (SGI bounce server)
8019	Apollo Domain
802E	Tymshare
802F	Tigan, Inc.
8035	Обратный протокол ARP
8036	Aeonic Systems
8038	DEC LANBridge
с 8039 по 803C	DEC (невыделенные)
803D	DEC Ethernet-шифрование
803E	DEC (невыделенный)
803F	Мониторинг LAN-трафика DEC
с 8040 по 8042	DEC (невыделенные)
8044	Planning Research Corp.
8046	AT&T

Шестнадцатеричное значение	Описание
8047	AT&T
8049	ExperData
805B	Stanford V Kernel exp.
805C	Stanford V Kernel prod
805D	Evans & Sutherland
8060	Little Machines
8062	Counterpoint Computers
8065	Univ. of Mass. @ Amherst
8066	Univ. of Mass. @ Amherst
8067	Veeco Integrated Auto.
8069	General Dynamics
8069	AT&T
806A	Autophon
806C	ComDesign
806D	Compulgraphic Corp.
с 806E по 8077	Landmark Graphics Corp.
807A	Matra
807B	Dansk Data Elektronik
807C	Merit Internodal
с 807D по 807F	Vitalink Communications
8080	Vitalink TransLAN III
с 8081 по 8083	Counterpoint Computers
809B	Appletalk
с 809C по 809E	Datability
809F	Spider Systems Ltd
80A3	Nixdorf Computers
с 80A4 по 80B3	Siemens Gammasonics Inc.
с 80C0 по 80C3	Кластер обмена данными DCA (DCA Data Exchange Cluster)
80C4	Banyan Systems
80C5	Banyan Systems
80C6	Pacer Software
80C7	Applitek Corporation
с 80C8 по 80CC	Intergraph Corporation
с 80CD по 80CE	Harris Corporation
с 80CF по 80D2	Taylor Instrument
с 80D3 по 80D4	Rosemount Corporation
80D5	SNA-служба IBM на основе Ethernet

Шестнадцатеричное значение	Описание
80CD	Varian Associates
с 80DE по 80DF	Integrated Solutions TRFS
с 80E0 по 80E3	Allen-Bradley
с 80E4 по 80FD	Datability
80F2	Retix
80F3	AppleTalk AARP (Kinetics)
с 80F4 по 80F5	Kinetics
80F7	Apollo Computer
с 80FF по 8103	Wellfleet Communications
с 8107 по 8109	Symbolics Private
8130	Hayes Microcomputers
8131	VG Laboratory Systems
с 8132 по 8136	Bridge Communications
с 8137 по 8138	Novell, Inc.
с 8139 по 813D	KTI
8148	Logicraft
8149	Network Computing Devices
814A	Alpha Micro
814C	SNMP
814D	BIIN
814E	BIIN
814F	Technically Elite Concept
8150	Rational Corp
с 8151 по 8153	Qualcomm
с 815C по 815E	Computer Protocol Pty Ltd
с 8164 по 8166	Charles River Data System
817D	XTP
817E	SGI/Time Warner prop.
8180	HIPPI-FP-инкапсуляция
8181	STP, HIPPI-ST
8182	Зарезарвировано для HIPPI-6400
8183	Зарезарвировано для HIPPI-6400
с 8184 по 818C	Silicon Graphics prop
818D	Motorola Computer
с 819A по 81A3	Qualcomm
81A4	ARAI Bunkichi
с 81A5 по 81AE	RAD Network Devices

Шестнадцатеричное значение	Описание
с 81B7 по 81B9	Xyplex
с 81CC по 81D5	Apricot Computers
с 81D6 по 81DD	Artisoft
с 81E8 по 81EF	Polygon
с 81F0 по 81F2	Comsat Labs
с 81F3 по 81F5	SAIC
с 81F6 по 81FB	VG Analytical
с 8203 по 8205	Quantum Software
с 8221 по 8222	Ascom Banking Systems
с 823E по 8240	Advanced Encryption System
с 827F по 8282	Athena Programming
с 8263 по 826A	Charles River Data System
с 829A по 829B	Inst Ind Info Tech
с 829C по 82AB	Taurus Controls
с 82AC по 8693	Walker Richer & Quinn
с 8694 по 869D	Idea Courier
с 869E по 86A1	Computer Network Tech
с 86A3 по 86AC	Gateway Communications
86DB	SECTRA
86DE	Delta Controls
86DD	IPV6
86DF	ATOMIC
с 86E0 по 86EF	Landis & Gyr Powers
с 8700 по 8710	Motorola
876B	Сжатие TCP/IP (RFC 1144)
876C	Автономные IP-системы (RFC 1701)
876D	Безопасные данные (RFC 1701)
880B	Протокол PPP
8847	Одноадресатное MPLS-вещание
8848	Многоадресатное MPLS-вещание
с 8A96 по 8A97	Invisible Software
9000	Обратная петля (Loopback)
9001	3Com(Bridge) XNS Sys Mgmt
9002	3Com(Bridge) TCP-IP Sys
9003	3Com(Bridge) обнаружение петель
FF00	BBN VITAL-LanBridge кэш
с FF00 по FF0F	ISC Bunker Ramo
FFFF	Зарезервирован (RFC 1701)

В этом приложении...

- **B.1: модуль системы обнаружения вторжений.** В разделе описываются этапы настройки и обслуживания *модуля системы обнаружения вторжений (Intrusion Detection System Module — IDSM)* как части системы безопасности.
- **B.2: модуль-анализатор коммутатора серии 6000.** В разделе описываются этапы конфигурирования, необходимые для использования *модуля анализа сети (Network Analysis Module — NAM)* с целью мониторинга данных от различных отправителей в коммутаторе.
- **B.3: модуль коммутирующей матрицы Catalyst 6000.** В этом разделе описывается *модуль коммутирующей матрицы (Switch Fabric Module — SFM)* и мониторинг его состояния.
- **B.4: модуль FlexWAN коммутатора Catalyst 6000.** В разделе описывается модуль FlexWAN и методика конфигурирования его интерфейсов. Упоминаемые в нем модули коммутаторов Catalyst описаны в следующих разделах книги:
  - **шлюз доступа (Access Gateway) коммутатора Catalyst 4000** описан в разделе "14.3: голосовые модули";
  - **восьми-портовый RJ-21 FXS-модуль коммутатора Catalyst 4000** описан в разделе "14.3: голосовые модули";
  - **восьми-портовый модуль голосовых каналов T1 и сдуй-0 коммутатора Catalyst 6000** описан в разделе "14.3: голосовые модули";
  - **24-портовый FXS-модуль коммутатора Catalyst 6000** описан в разделе "14.3: голосовые модули";
  - **модуль коммутации по содержимому коммутатора Catalyst 6000** (Content Switching Module) описан в главе 10, "Балансирование нагрузки по серверу (SLB)".

Приложение B

Дополнительные модули коммутаторов Catalyst

B.1: модуль системы обнаружения вторжений

- Модуль IDSM способен анализировать поток данных и оповещать вторжение или аномальную активность на основании набора сигнатур (signature) атак.
- Модуль IDSM может осуществлять мониторинг данных на скорости до 100 Мбит/с. Если скорость передачи данных выше этого предела, то некоторые пакеты будут потеряны и атака может пройти неопознанной.
- Пользователи могут подписаться на бюллетень IDS, в котором описываются обновления сигнатур, пакеты исправлений (service packs) или новости о продуктах. Для подписки необходимо ввести свой адрес электронной почты и пароль на странице `www.cisco.com/warp/public/779/largeent/it/ids_news/subscribe.html`.
- В модуле IDSM имеются два логических порта коммутатора: порт 1 — порт перехвата или "прослушивания" (sniffing), и порт управления (порт 2)
- Изучаемый трафик должен отправляться на порт перехвата (capture port). Мониторинг возможен только для трафика той сети, членом которой является порт перехвата. Порт перехвата может быть магистральным (trunk), что позволяет осуществлять мониторинг трафика всех VLAN-сетей на магистральном канале
- Модуль IDSM управляется внешним приложением. При обнаружении атаки модуль отправляет оповещение административному приложению. Таким приложением может быть один из программных продуктов *Cisco Secure Director*, *Cisco Secure Policy Manager* или *HP OpenView*.
- В одно шасси Catalyst 6000 можно установить несколько IDSM-модулей. Все модули можно настроить на мониторинг отдельных потоков данных до 100 Мбит/с каждый.

Конфигурирование модуля

1. Доступ к модулю IDSM.

а) Установление CLI-сеанса с модулем IDSM.

Система COS	`session` *module*
Система IOS	`session slot` *module* `processor 1`

С помощью этих команд инициализируется CLI-сеанс с модулем IDSM, установленным в гнезде с номером *module*. Для закрытия сеанса необходимо ввести последовательность символов <^>+<D> или <CTRL>+<D> либо команду `exit`.

б) Регистрация с правами администратора.

Модуль IDSM	`login username` *username* `password` *password*

Параметр *username* — пользовательское имя администратора (текстовая строка; по умолчанию "cisconds"). Параметр *password* — пароль (текстовая строка; по умолчанию "attack").

в) Изменение пароля администратора (*необязательно*).

Модуль IDSM	`configure terminal`
	`password`
	(режим глобальной конфигурации)

Стандартные установки пароля администратора рекомендуется изменить. Паролем является текстовая строка длиной до 15 символов.

г) Первоначальное конфигурирование модуля IDSM (*необязательно*).

Модуль IDSM	`setup`

После перехода в режим настройки (setup mode) модуль IDSM затребует все необходимые параметры. Ниже перечислены параметры, необходимые как для локального IDS-сенсора (IDS sensor), так и для удаленного IDS-диспетчера (IDS director)

- IP-адрес, маска подсети и стандартный шлюз
- Имя узла (до 256 символов), идентификатор узла (от 1 до 65535), а также UDP порт POP протокола (Post Office Protocol — от 256 до 65 535; стандартно — 45 000). Идентификатор (ID) узла представляет собой число, которое уникально идентифицирует IDSM-модуль среди других IDS-сенсоров в организации
- Название организации (до 256 символов и ее ID (от 1 до 65 535). Внутри IDS-домена сенсоры и их управляющие диспетчеры должны иметь общие имя и идентификатор организации.

д) Проверка сетевой связи с модулем IDSM (*необязательно*).

- Ping-запрос к удаленному узлу

Модуль IDSM	`diagnostics` `ping ip-address`

- Трассировка маршрута к удаленному узлу.

Модуль IDSM	`diagnostics` `traceroute` *ip_address*

2. Отключение модуля IDSM (*необязательно*).

Внимание!

Прежде чем удалять модуль IDSM из шасси коммутатора, необходимо его соответствующим образом отключить. Чтобы инициировать корректное завершение работы, необходимо воспользоваться одним из описанных ниже методов и дождаться завершения процесса отключения. О полном отключении свидетельствует изменение цвета светодиодного индикатора с зеленого на желтый или его отключение. После этого можно безопасно демонтировать модуль.

а) Отключение модуля из CLI-интерфейса IDSM.

Модуль IDSM	`shutdown`

б) Отключение или перезапуск модуля из CLI-интерфейса (*необязательно*)

Система COS	`set module shutdown` *module* или `reset` *module*
Система IOS	`hw-module module shutdown` *module* или `hw-module module` *module* `reset`

С помощью ключевого слова `shutdown` можно отключить IDSM-модуль, установленный в гнездо с номером *module*. Если после этого перезагрузить или отключить, а затем включить питание коммутатора, то модуль IDSM также перезагрузится. Чтобы повторно подключить модуль IDSM к обслуживанию, нужно использовать ключевое слово `reset`.

в) Отключение или восстановление питания модуля IDSM (*необязательно*).

Система COS	`set module power down` *module* или `set module power up` *module*
Система IOS	`no power enable module` *module* или `power enable module` *module*

г) Использование в качестве последнего средства кнопки отключения модуля IDSM (*необязательно*)

Чтобы нажать кнопку отключения на передней панели модуля IDSM, можно использовать небольшой остроконечный предмет, например, конец канцелярской скрепки. Кнопка расположена с правой стороны от светодиодного индикатора состояния.

3. Команды для активации в модуле IDSM функции обнаружения вторжения.

Из интерфейса административного приложения, такого, как Cisco Secure Director, можно выбрать набор сигнатур атак, который будет использоваться модулем

IDSM. Для получения более подробной информации рекомендуется изучить документацию по административному приложению.

4. Определение отправителя трафика для мониторинга (*необязательно*).

   a) Выбор интересующего трафика (*необязательно*).

   Интересующий администратора трафик можно выбрать с помощью списка доступа VLAN-сети (*VLAN access control list — VACL*).

Система COS	`set security acl ip` *acl-name* `{permit | deny [re-direct` *'adj name | mod/port}] protocol src-ip-spec dest-ip-spec* `[precedence` *precedence*`] [tos` *tos*`] [fragment] capture [before` *editbuffer index* `| modify` *editbuffer index*`] [log]`

Система IOS	`access-template` *{acl number | acl name} {temporary-list-name} src-ip-spec dest-ip-spec* `[timeout` *minutes*`]`
	`vlan access-map` *name {seq#}*
	(режим глобальной конфигурации)
	`match {ip address` *{acl-number | acl-name}*`}`

Для трафика, который будет перехватываться или отслеживаться модулем IDSM, следует добавить оба ключевых слова `permit` и `capture`. Для остального трафика, который будет пересылаться без передачи, следует добавить только ключевое слово `permit`. Трафик, который не перенаправляется, отмечается с помощью ключевого слова `deny`. Следует позаботиться о том, чтобы в конце VACL-списка присутствовала строка `permit ip any any`, которая позволяет пересылать весь остальной неуказанный трафик.

В простейшем случае весь трафик VLAN сети перехватывается с помощью команды `permit ip any any capture`.

Полная информация о конфигурировании VACL-списков приведена в разделе "11.4: списки доступа VLAN-сетей".

- Включение VACL-списка в аппаратном обеспечении

Система COS	`commit security acl {`*acl-name* `| all}`
Система IOS	Нет

COS-коммутатор должен сначала скомпилировать VACL-список и загрузить его в коммутирующее аппаратное обеспечение. Можно скомпилировать и ввести один VACL-список (*acl name*) или все сконфигурированные списки (`all`).

- Копирование интересующего трафика в IDSM-модуль

Система COS	`set security acl capture-ports` *mod/1*
Система IOS	`action forward capture {`*interface slot/1*`}`

Трафик, который совпадает с VACL-списком, копируется в порт перехвата, который определяется номером слота IDSM-модуля и номером порта (1).

- Применение VACL-списка к одной или нескольким VLAN-сетям.

Система COS	`set security acl map acl-name vlan`
Система IOS	`vlan filter map-name vlan-list vlan-list` (режим глобальной конфигурации)

Список доступа VLAN-сети с указанным именем (*acl-name* — текстовая строка длиной до 32 символов) применяется к изучаемому трафику во VLAN сети, определенной номером (`vlan` — 1–1005 или 1025–4094).

- Измерение количества отслеживаемого трафика.

  Следует убедиться, что полоса пропускания, которая потребляется перехватываемым или отслеживаемым трафиком, не достигает уровня 100 Мбит/с. При соблюдении этого условия модуль IDSM способен изучить все пакеты в каждом потоке данных.

  Можно сконфигурировать дополнительный VACL-перехват для неиспользуемого гигабитного Ethernet-интерфейса. Данные, копируемые в порт перехвата модуля IDSM, также копируются в неиспользуемый порт коммутатора. Впоследствии можно определить количество перехваченных данных с помощью команды `show mac mod/port` или `show interface`.

Система COS	`set security acl capture-ports` *mod/port*
Система IOS	`action forward capture /interface` *slot/number*

6) Мониторинг SPAN-отправителя.

Совет

Полная информация по конфигурированию *анализатора коммутируемых портов (Switched Port Analyzer — SPAN)* или службы мониторинга портов приведена в разделе "2.3. анализатор коммутируемых портов".

- Мониторинг порта коммутатора (*необязательно*).

Система COS	`set span` *src mod/src ports* `dest mod/1 [rx | tx | both] (inpkts {enable | disable}) [learning {enable | disable}] [multicast {enable dis- able}] [filter vlans, ..] [create]`

Система IOS	`monitor session` *session* `source interface type number [rx tx | both]`
	`monitor session session destination interface type 1`
	`monitor session session filter vlan vlans`
	(все команды вводятся в режиме глобальной конфигурации)

Отправитель данных идентифицируется как определенный порт коммутатора. Получателем является порт 1 модуля IDSM. Фильтрация трафика определенных VLAN-сетей с целью мониторинга осуществляется при помощи ключевого слова `filter`.

- Мониторинг VLAN-сети (*необязательно*).

Система COS	`set span src-vlans dest-mod/1 [rx | tx both] (inpkts {enable | disable}) [learning {enable | disable}] [multicast {enable | disable}] [create]`

Система IOS	`monitor session` *session* `source vlan` *vlans* `[rx | tx | both]` `monitor session` *session* `destination interface` *type 1* (обе команды вводятся в режиме глобальной конфигурации)

В качестве источника изучаемого трафика может быть задана одна или несколько VLAN-сетей. Получателем трафика является порт 1 модуля IDSM.

- **Измерение количества отслеживаемого трафика.**

  Необходимо убедиться, что полоса пропускания, которая потребляется перехватываемым или изучаемым трафиком, определенным в ходе этапов 4.б и 4.в, не достигает уровня 100 Мбит/с. При соблюдении этого условия модуль IDSM способен изучить все пакеты в каждом потоке данных.

  Простейшим способом контроля скорости трафика к модулю IDSM является выбор порта-отправителя с полосой пропускания 100 Мбит/с.

  Можно также использовать команду `show top -n` для отображения наиболее используемых портов коммутатора.

5. Обновление программного обеспечения модуля IDSM (*необязательно*).

   а) Выбор активного раздела жесткого диска

Система COS	`set boot device hdd:`*partition-num*
Система IOS	Нет

Модуль IDSM в числе с номером *slot* использует образ программного обеспечения в разделе (*partition-num*) 1 (стандартный раздел приложений) или 2 (раздел обслуживания). После того как модуль IDSM запускает образ в активном разделе, появляется возможность обновить другой раздел.

   б) Установка кэшированного образа из неактивного раздела (*необязательно*).

- Переход в режим диагностики.

Модуль IDSM	`diag`

- Приведенная ниже команда позволяет убедиться, что кэшируется необходимый образ.

Модуль IDSM	`ids installer system /cache /show`

- Установка кэшированного образа.

Модуль IDSM	`ids-installer system /cache /install`

- Перезагрузка модуля IDSM.

Система COS	`reset` *module* `hdd:2`
Система IOS	`hw-module module` *mod#* `reset hdd:2` (режим глобальной конфигурации)

Модуль IDSM перезагружается, используя образ в разделе приложений (раздел 1).

в) Установка образа с FTP-сервера (*необязательно*).

- Переход в режим диагностики.

Модуль IDSM	`diag`

- Указание IP-адресов для процесса обновления (*необязательно*).

Модуль IDSM	`ids-installer netconfig /configure /ip=`*ip-addr* `/subnetmask /gwaddr=`*ip_addr*

При обновлении разделы приложений с FTP-сервера модуля IDSM используют определенный IP-адрес (*ip-addr*), маску подсети и адрес шлюза (*gw-ip_addr*).

- Загрузка образа с FTP-сервера

Модуль IDSM	`ids-installer system /nw /install /server=`*ip-addr* `/user=`*account* `/save={yes no} /dir=`*ftp-path* `/prefix=`*file-prefix*

IDSM-модуль связывается с FTP-сервером (*ip-addr*), используя регистрационное имя пользователя (*account* — текстовая строка). После загрузки с сервера образа в его установки с помощью ключевого слова `yes` также можно кэшировать образ в IDSM-модуле. Образ находится в указанном FTP-каталоге (*ftp-path*). Имя файла образа задается параметром *file prefix* (текстовая строка), не включая расширения "*.dat*".

- Перезагрузка модуля IDSM.

Система COS	`reset -cdisk hdd:1`

Модуль IDSM перезагружается, используя образ в разделе приложений (раздел 1).

6. Обновление программного обеспечения модуля IDSM с помощью пакетов исправлений (*необязательно*)

а) **Проверка** активной версии (*необязательно*)

Модуль IDSM	`show config`

б) Загрузка пакета исправлений (service pack) на FTP-сервер.

Совет

Найти пакеты исправлений для модуля IDSM можно в Центре программного обеспечения (Software Center) на Web-узле Cisco.com *www.cisco.com/kobayashi/sw-center/sw-ciscosecure.shtml* в разделе "Cisco Intrusion Detection System (IDS)". Программное обеспечение для IDS-модуля Catalyst 6000 доступно по ссылке "Latest Software". (Для доступа к ресурсам Центра программного обеспечения необходима учетная запись пользователя CCO, а также действующий контракт на обслуживание.)

в) Применение пакета исправлений.

Модуль IDSM	`configure terminal` `apply servicepack site ftp://`*ip-addr* `user` *account* `dir` *path* `file` *filename*

Модуль IDSM связывается с FTP-сервером (IP-адрес — *ftp-ip-addr*), используя указанное имя пользователя (*account* — текстовая строка). Пакет исправлений хранится в каталоге *path*, имя файла задаётся параметром *filename* (текстовая строка, включая расширение *exe*).

Последний примененный пакет исправлений можно удалить с помощью команды `remove servicepack`.

7. Обновление базы данных IDS-сигнатур (*необязательно*).

   а) Проверка активной версии (*необязательно*).

Модуль IDSM	`show config`

   б) Загрузка базы данных сигнатур на FTP-сервер

Совет

Найти базы данных сигнатур для модуля IDSM или их обновления можно в Центре программного обеспечения (Software Center) на Web-узле Cisco.com: `www.cisco.com/kobayashi/sw-center/sw-cisco-secure.shtml` в разделе "Cisco Intrusion Detection System (IDS)". Программное обеспечение для IDS-модуля Catalyst 6000 доступно по ссылке "Latest Software". (Для доступа к ресурсам Центра программного обеспечения необходима учетная запись пользователя CCO, а также действующий контракт на обслуживание.)

   в) Обновление сигнатуры

Модуль IDSM	`configure terminal` `apply signatureupdate write ftp-ip-addr user account dir path file filename`

IDSM связывается с FTP-сервером (IP-адрес — *ftp-ip-addr*), используя указанное имя пользователя (*account* — текстовая строка). Пакет обновления сигнатур хранится в каталоге *path*, имя файла задается параметром *filename* (текстовая строка, включая расширение *exe*).

Последний примененный пакет обновлений сигнатур можно удалить с помощью команды `remove signatureupdate`.

Отображение информации о модуле IDSM

В табл. B.1 перечислены команды коммутатора, с помощью которых можно отобразить полезные сведения о модуле IDSM.

Таблица B.1. Команды для отображения информации о модуле IDSM

Функция отображения	Операционная система коммутатора	Команда
Версии IDSM	IDSM	`show version`
IDSM-конфигурация	IDSM	`show configuration`
VACL-списки для перехвата	COS	`show security acl info all`

Функция отображения	Операционная система коммутатора	Команда
	IOS	`show vlan filter [{access-map map-name} | {vlan vlan-id}]`
VACL-списки, назначенные VLAN-сетям	COS	`show security acl map {acl name | vlan | all}`
	IOS	`show vlan access-map [map-name]`
Активные SPAN-сеансы	COS	`show span`
	IOS	`show port monitor`

B.2: модуль-анализатор сети коммутатора серии 6000

- Модуль NAM (Network Analysis Module — модуль анализа сети) исследует потоки данных на коммутаторе и обеспечивает аналитические функции с использованием удаленного мониторинга (Remote Monitoring — RMON), функции RMON2, а также MIB объектов протокола SNMP (Simple Network Management Protocol — простой протокол управления сетью).
- С помощью описываемого модуля можно исследовать трафик, поступающий от VLAN-сети или SPAN-порта коммутатора.
- Модуль NAM позволяет накапливать и анализировать NDE-информацию (*Netflow data export — функция экспорта данных Netflow*).
- Управление модулем может осуществляться с помощью перечисленных ниже приложений:
  - NAM Traffic Analyzer — внутреннее Web-приложение;
  - CiscoWorks TrafficDirector;
  - NetScout nGenius Real-Time Monitor (RTM);
  - CiscoWorks2000 (по протоколу SNMP).

Конфигурирование модуля

1. Доступ к NAM-модулю.

   а) Назначение административного NAM-порта VLAN-сети

Системы COS	Нет

Системы IOS	`interface gigabitethernet mod/1`
	(режим глобальной конфигурации)
	`switchport access vlan vlan-number`
	(режим конфигурирования интерфейса)

В IOS-коммутаторе административный порт NAM-модуля (NAM management port) порт 1 необходимо назначить сети VLAN-порту, чтобы можно было коммутировать NAM-трафик, направленный административному приложению и от него.

Для COS коммутатора это не требуется. Административный порт NAM-модуля автоматически назначается VLAN-сети, связанной с sc0-интерфейсом коммутатора.

б) Установление CLI-сеанса с модулем NAM.

Система COS	`session module`
Система IOS	`session slot module processor 1`

С помощью этих команд инициируется сеанс командной строки с модулем NAM, установленным в слоте *module*. Для закрытия сеанса следует воспользоваться комбинацией клавиш <^a+d> или <Ctrl>+d>.

в) Вход в систему с правами администратора.

Модуль NAM	`login: root` `Password:` *password*

В команде используется имя пользователя `root` и пароль (*password* — текстовая строка; стандартно — "cisco").

г) Изменение пароля администратора (*необязательно*)

Модуль NAM	`password root`

Стандартное значение пароля администратора рекомендуется изменить. Пароль представляет собой текстовую строку длиной до 15 символов.

д) Первоначальное конфигурирование IP-параметров (*необязательно*).

- Настройка IP-адреса

Модуль NAM	`ip address` *ip-address subnet-mask*

Для управления модулем NAM используется его IP-адрес. Следует убедиться, что IP-адрес выбран соответствующим образом для совпадения с VLAN административного порта.

- Настройка широковещательного адреса

Модуль NAM	`ip broadcast` *broadcast-address*

Широковещательный адрес следует устанавливать согласно параметрам IP-сети и маски подсети административного порта NAM-модуля.

- Установка имени узла и имени домена.

Модуль NAM	`ip host` *name* `ip domain` *domain-name*

Модуль NAM идентифицирует себя по имени узла и доменному имени (текстовые строки).

- Настройка адреса стандартного шлюза.

Модуль NAM	`ip gateway default-gateway`

С помощью этой команды указывается IP-адрес стандартного шлюза в локальной сети административного порта модуля NAM.

- Указание одного или нескольких имён серверов.

Модуль NAM	`ip nameserver ip address [ip address ...]`

г) Первоначальное конфигурирование параметров протокола SNMP

- Указание местоположения системы.

Модуль NAM	`snmp location location`

Местоположение системы (*location* — текстовая строка) используется для описания места физического расположения модуля NAM.

- Указание контактов лиц, поддерживающих систему

Модуль NAM	`snmp contact contact`

В команде определяются имя лица или группы лиц (параметр *contact* — текстовая строка), с которыми следует контактировать по вопросам обслуживания модуля NAM

- Указание имени системы

Модуль NAM	`snmp name name`

Модуль NAM идентифицируется в SNMP-запросе по имени (*name* — текстовая строка).

- Указание строки сообщества с правами чтения-записи (*необязательно*).

Модуль NAM	`snmp community community-string rw`

Административное SNMP приложение может считывать или записывать MIB-переменные, только если в нем используется строка сообщества (*community-string* — текстовая строка), которая задана в NAM-модуле

- Указание строки сообщества только для чтения.

Модуль NAM	`snmp community community-string ro`

Административное SNMP-приложение может только считывать MIB-переменные, если в нем используется строка сообщества (*community-string* — текстовая строка), которая задана в NAM-модуле.

ж) Проверка сетевой связи с NAM-модулем (*необязательно*).

- Ping-запрос к удаленному узлу.

Модуль NAM	`ping [-mv] [-c count] [-i wait] [-p pattern] [-s packetsize] {hostname	ip address}`

Эхо-пакеты протокола **ICMP** отправляются удаленному узлу с указанным именем (*hostname* — текстовая строка) или IP-адресом (*ip-address*).

Чтобы вместо имен узлов отобразить адреса в числовом виде, нужно использовать флаг `-n`. Для получения более подробных сведений используется флаг `-v`. Количество пакетов может быть задано с помощью инструкции `-c count`. При использовании флага `-i` возникает пауза между отправками эхо-пакетов, длительность которой задается с помощью параметра *wait*. Эхо-пакеты могут заполняться структурами (*pattern* — до 16 байтов в шестнадцатеричном виде). Для этого используется флаг `-p`. С помощью флага `-s` можно установить размер эхо-пакетов (параметр *packetsize*).

- Трассировка маршрута к удаленному узлу.

Модуль NAM	`traceroute [-Inv] [-f first-ttl] [-m max-ttl] [-p port] [-s source-addr] [-t tos] [-w wait-time] {hostname | ip-address}`

Пакеты запросов трассировки маршрута отправляются удаленному узлу, который указывается с помощью параметра *hostname* (текстовая строка) или *ip-address*. Если не задан флаг `-I`, приводящий к отправке ICMP эхо-пакетов, используются UDP-запросы. Для отображения адреса в числовом виде вместо имени узлов используется флаг `-n`. Для получения более подробных сведений используется флаг `-v`. Первоначальное и максимальное время жизни (количество транзитных переходов) задается с помощью флагов `-f` и `-m` соответственно.

Номер используемого UDP-порта задается с помощью параметра `-p`. Адрес отправителя, отличный от административного порта модуля NAM, можно указать с помощью флага `-s`. Значение *типа обслуживания (Type of Service — ToS)* можно задать с помощью флага `-t`. При использовании параметра `-w` модуль NAM ожидает ответа на пробу в течение *wait* секунд.

2. Отключение модуля NAM (*необязательно*)

Внимание!

Прежде чем удалять модуль NAM из шасси коммутатора, необходимо соответствующим образом отключить его. Чтобы инициировать корректное завершение работы, используйте один из описанных ниже методов и дождитесь завершения процесса отключения. О полном отключении свидетельствует изменение цвета светодиодного индикатора с зеленого на желтый или его отключение. После этого можно безопасно демонтировать модуль.

а) Отключение модуля из интерфейса командной строки NAM

Модуль NAM	`shutdown`

б) Отключение или перезагрузка модуля из командной строки коммутатора (*необязательно*).

Система COS	`set module shutdown` *module* или `reset` *module*
Система IOS	`hw-module` *module* `shutdown` *module* или `hw-module` *module module* `reset`

С помощью ключевого слова shutdown можно отключить NAM-модуль, установленный в гнезде *mod/c*. Если после этого перезагрузить или отключить, а затем включить питание коммутатора, то модуль NAM также перезагрузится. Для повторного подключения модуля NAM к обслуживанию используется ключевое слово reset.

в) Отключение или восстановление питания модуля NAM (*необязательно*).

Система COS	`set module power down` *mod/c* или `set module power up` *mod/c*
Система IOS	`no power enable module` *mod/c* или `power enable module` *mod/c* (режим глобальной конфигурации)

г) Использование в качестве последнего средства кнопки отключения модуля NAM (*необязательно*).

Чтобы нажать кнопку отключения на передней панели модуля NAM, можно использовать небольшой остроконечный предмет, например, канцелярскую скрепку. Кнопка расположена справа от светодиодного индикатора состояния.

3. Выбор отправителя трафика для мониторинга.

а) Мониторинг NDE-информации (*необязательно*).

Совет

Модуль NAM можно настроить на сбор NDE-информации от одного или нескольких источников. Чтобы в локальном коммутаторе, содержащем модуль NAM, активизировать функцию NDE-отправителя, необходимо выполнить описанные ниже инструкции. Если локальный коммутатор является исключительно COS-устройством, необходимо также настроить функциональную плату многоуровневого коммутатора (Multilayer Switch Feature Card — MSFC/MSFC2), с помощью которой генерируется NDE-информация. Полные сведения по конфигурированию этой функции приведены в разделе "8.3: экспорт данных NetFlow".

- Включение функции NDE на PFC-плате.

Система COS	`set snmp extendedrmon netflow {enable | disable}` *mod*
	`set mls nde {enable | disable}`
	`set mls nde version {1 | 7 | 8}`

Система IOS	`mls nde sender [version` *version*`]` (режим глобальной конфигурации)

Для платы PFC/PFC2 (Policy Feature Card — функциональная плата политик) трафик, коммутируемый на третьем уровне, может учитываться как данные NDE версии 7. В COS-коммутаторе плату PFC можно включить для отправки NDE-данных NAM-модулю, расположенному в гнезде *mod*.

- Включение NDE на интерфейсах третьего уровня (*необязательно*).

Для платы MSFC/MSFC2 маршрутизируемый трафик может регистрироваться как данные NDE версии 1, 5 или 6. Функцию NDE необходимо

включать на всех интерфейсах третьего уровня, где трафик маршрутизируется с помощью платы MSFC.

Система COS	Нет
Система IOS	`ip route-cache flow` (режим конфигурирования интерфейса)

- Идентификация NDE отправителя.

Система COS	Нет
Система IOS	`ip flow-export source {{interface interface-number} | {null interface-number} | {port-channel number} | {vlan vlan id}}` (режим глобальной конфигурации)

NDE-пакеты отправляются с указанным адресом отправителя. В качестве отправителя может использоваться либо интерфейс (Ethernet, null (несуществующий логический интерфейс) или EtherChannel), либо VLAN сеть.

- Идентификация NDE-коллектора.

Система COS	`set mls nde ip-address 3000`

Система IOS	`mls np nde-address ip-address`
	`ip flow-export destination ip address 3000`
	(обе команды вводятся в режиме глобальной конфигурации)

Модуль NAM с IP-адресом *ip address* будет получать NDE-информацию через UDP-порт 3000, который является обязательным для NAM-модуля.

- Указание маски потока, используемой для генерации netflow-данных (*необязательно*).

Система COS	Нет
Система IOS	`mls flow {ip {destination | destination-source | full} | {ipx {destination | destination-source}}`

Функция NDE отправляет сведения о потоке на основании адреса получателя (`destination`, стандартная настройка) и отправителя (`destination-source`) или на основании адресов и портов отправителя и получателя (`full`).

- Использование фильтров для создания отчетов об определенных потоках данных (*необязательно*).

Система COS	`set mls nde flow {exclude | include} [destination ip-address mask] [source ip-address mask] [protocol protocol] [src-port number] [dst-port number]`
Система IOS	`mls nde flow {exclude | include} {destination ip-address mask | source ip-address mask {dest-port number | src-port number}}` (режим глобальной конфигурации)

С помощью функции NDE экспортируются потоки, которые соответствуют (include) или не соответствуют (exclude) остальным параметрам фильтра. Фильтрация может осуществляться на основании IP-адреса получателя (destination), номера порта получателя (dest-port), IP-адреса отправителя (source) и номера порта отправителя (src-port).

б) Мониторинг SPAN-сеанса (*необязательно*).

- Мониторинг порта коммутатора (*необязательно*).

Система COS	`set span` *src-mod/src-ports anal-mod/1* `{rx | tx | both} [inpkts {enable | disable}] [learning {enable | disable}] [multicast {enable | disable}] [filter vlans...] [create]`
Система IOS	`monitor session` *session* `source interface type num-ber {rx | tx | both}` `monitor session` *session* `destination interface gigabitethernet` *mod/1* `monitor session` *session* `filter vlan` *vlans* (все команды вводятся в режиме глобальной конфигурации)

Отправитель трафика идентифицируется как определенный порт коммутатора. Получателем является NAM-порт 1. С помощью ключевого слова `filter` можно осуществлять фильтрацию определенных VLAN-сетей трафика с целью мониторинга.

- Мониторинг VLAN-сети (*необязательно*)

Система COS	`set span` *src-vlans dest-mod/1* `[rx | tx | both] [inpkts {enable | disable}] [learning {enable | disable}] [multicast {enable | disable}] [create]`
Система IOS	`monitor session` *session* `source vlan` *vlans* `{rx | tx | both}` `monitor session` *session* `destination interface gigabitethernet` *mod/1* (все команды вводятся в режиме глобальной конфигурации)

В качестве отправителей, трафик которых будет отслеживаться, можно задать одну или несколько VLAN-сетей. Получателем является NAM-порт 1.

4. Обновление программного обеспечения NAM-модуля (*необязательно*)

а) Переключение обновляемого раздела в неактивное состояние.

Система COS	`reset` *module hdd:partition*
Система IOS	`hw-mod module` *module* `reset hdd:partition` (режим глобальной конфигурации)

Для обновления раздела приложений (1) следует перезагрузить NAM-модуль и запустить с раздела обслуживания (параметр *partition*, 2). Напротив, если необходимо обновить программное обеспечение раздела обслуживания, следует запустить систему с раздела приложений (*partition* 1).

б) Установка образа с FTP-сервера.

Модуль NAM	**upgrade** *ftp-url*

NAM-модуль способен загружать программное обеспечение с любого FTP-сервера (включая сервер CCO по адресу ftp.cisco.com), связь с которым возможна посредством данной сети. Прежде всего необходимо убедится, что сервер отвечает на ping-запросы.

Местонахождение программного обеспечения определяется с помощью URL-адреса (параметр *ftp-url*). Для FTP-сервера, поддерживающего анонимный доступ, используется формат ftp://host/absolute-path/filename. В противном случае следует указать имя пользователя и пароль, используя формат ftp://user@hostname/absolute-path/filename.

в) Перезагрузка NAM-модуля в раздел приложений.

Система COS	**reset** *module* **hdd:1**
Система IOS	**hw-mod module** *module* **reset hdd:1** (режим глобальной конфигурации)

5. Применение исправлений программного обеспечения в модуле NAM (*необязательно*).

а) Следует убедиться, что NAM использует образ приложения.

б) Загрузка и применение исправления (patch).

Модуль NAM	**patch** *ftp-url*

NAM-модуль способен загружать исправления программного обеспечения с любого FTP-сервера (включая сервер CCO по адресу ftp.cisco.com), связь с которым возможна посредством имеющейся сети. Прежде всего необходимо убедиться, что сервер отвечает на ping-запросы.

Местонахождение исправления (patch) определяется с помощью URL-адреса (параметр *ftp-url*). Для FTP-сервера, поддерживающего анонимный доступ, используется формат ftp://host/absolute-path/filename. В противном случае следует указать имя пользователя и пароль, используя формат ftp://user@hostname/absolute-path/filename.

6. Доступ к HTTP-серверу модуля NAM (*необязательно*).

а) Установка исправления для реализации строгого шифрования (3DES).

Модуль NAM	**patch** *ftp-url*

Прежде всего в модуле NAM необходимо применить исправление, которое активизирует криптографические функции. Найти это исправление можно на FTP-сайте корпорации Cisco по адресу ftp://ftp.cisco.com/cisco/crypto/3DES/lan/catalyst/6000/nam/c6svc-nam-strong-crypto-patchK9-1.0-1.1386.rpm.

б) Выбор безопасного HTTP-порта.

Модуль NAM	**ip http secure port** *port*

TCP-порт, используемый для безопасного HTTP-доступа, задается параметром *port* (1–65 535). Как правило, используется порт 8080.

в) Включение HTTP-сервера NAM-модуля.

Модуль NAM	`ip http server enable`

г) Использование собственных (self-signed) сертификатов для безопасных соединений с сервером *(необязательно)*.

Собственные сертификаты генерируются модулем NAM и передаются HTTP-клиентам для безопасных HTTP-сеансов.

д) Использование для безопасных серверных соединений *CA-сертификата (необязательно)*.

- Запрос сертификата от CA *(Certificate Authority — бюро сертификации)*.

Модуль NAM	`ip http secure generate certificate-request`

При выполнении команда запрашивает у пользователя необходимые сведения об организации и сервере. Затем запрос сертификата отображается в виде нескольких строк символов. Для того чтобы действительно выполнить запрос сертификата, необходимо скопировать информацию запроса и вставить ее в сообщение запроса, которое вручную отправляется CA. От CA должен поступить ответ с данными сертификата.

- Установка сертификата.

Модуль NAM	`ip http secure install certificate`

Команда запрашивает у пользователя сертификационные сведения. Необходимо скопировать и вставить строки символов сертификата, включая строки, которыми обозначаются его начало и конец. Ввод данных сертификата завершается вводом символа точки и нажатием клавиши <Enter>.

Отображение информации о NAM-модуле

В табл. В.2 перечислены команды коммутатора, которые можно использовать для получения различной полезной информации о модуле NAM.

Таблица В.2. Команды для отображения NAM-информации

Функция отображения	Операционная система коммутатора	Команда
SNMP-конфигурация	NAM	`show snmp`
IP-конфигурация	NAM	`show ip`
Использование процессора	NAM	`show cpu`
Использование памяти	NAM	`show memory`
Версия и серийный номер	NAM	`show bios`
HTTP-сертификаты	NAM	`show certificate` `show certificate-request`
Установленные функции	NAM	`show patches` `show options`

B.3: модуль коммутирующей матрицы Catalyst 6000

- Модуль SFM (Switch Fabric Module – модуль коммутирующей матрицы) обеспечивает перекрестный механизм коммутации (crossbar switching fabric) на скорости 256 Гбит/с. Модуль выделяет два порта со скоростью 8 Гбит/с для каждого модуля в шасси коммутатора Catalyst 6500 (каждый порт функционирует в полудуплексном режиме 16 Гбит/с).
- Модули коммутационной матрицы (Fabric-enabled modules) могут подключаться только к одному из двух выделенных SFM-портов. Коммутационные модули (Fabric-only modules) способны подключаться к обоим SFM-портам.
- Модель SFM обеспечивает несколько режимов связи между модулями. Эти режимы описаны ниже.
  - **Сквозной режим (flow-through mode)** используется при установленных модулях, не поддерживающих коммутационную матрицу. Данные проходят через шины обычной объединительной платы.
  - **Усеченный режим (truncated mode)** используется в ситуации, когда данными обмениваются модули коммутационной матрицы, но установлено несколько модулей, не поддерживающих ее. Данные проходят через канал матрицы, если модуль, не поддерживающий ее, не задействован.
  - **Компактный режим (compact mode)** используется в случае, если в коммутаторе установлены только модули коммутационной матрицы. Данные проходят через канал матрицы в виде компактной версии, в результате чего достигается наилучшая производительность коммутации.
- В целях резервирования в коммутатор можно устанавливать два SFM-модуля. Модуль SFM, установленный в верхнее гнездо, функционирует в качестве основного, а модуль SFM в нижнем гнезде – в качестве дополнительного.
- Структура коммутаторов семейства Catalyst 6000 более подробно описана в главе 2, "Функции коммутатора".

Конфигурирование модуля

1. Выбор режима коммутации (*необязательно*)

Система COS	`set system switchmode allow {truncated | bus-only}`
Система IOS	Нет

Коммутатор допускает использование усеченного режима (`truncated`, стандартная настройка), если установлено малое число модулей, не поддерживающих коммутационную матрицу. Если все модули поддерживают матрицу, то модуль SFM может войти в компактный режим. При использовании ключевого слова `bus-only` SFM-модуль переводится в сквозной режим.

2. Выбор резервного (fallback) режима (*необязательно*).

Если модуль SFM выходит из строя, коммутатор может начать использовать обычную системную шину (стандартный режим `bus-mode`). Если использова-

лось ключевое слово none, то во время сбоя SFM-модуля коммутация трафика осуществляться не будет.

Система COS	`set system crossbar-fallback {bus-mode | none}`
Система IOS	Нет

3. **Изменение логотипа на жидкокристаллическом индикаторе** (*необязательно*).

Система COS	`set banner lcd d message d`
Система IOS	`fabric led-banner d message d` (режим глобальной конфигурации)

Обычно на LCD-индикаторе SFM-модуля отображаются логотип и название корпорации Cisco Systems. Можно ввести сообщение (текстовую строку длиной до 80 символов), которое начинается и заканчивается между ограничивающими символами *d*. Эти символы не могут входить в состав сообщения. Сообщение можно ввести в виде одной или нескольких строк текста. Внутрь сообщения можно ввести маркеры, которые отображают их текущие значения. Вместо маркера *$(hostname)* может быть подставлено имя узла коммутатора, а вместо маркера *$(domain)* — его доменное имя.

Отображение информации о модуле SFM

В табл. В.3 перечислены команды коммутатора, которые можно использовать для получения полезных сведений о работе модуля SFM.

Таблица В.3. Команды для отображения SFM-информации

Функция отображения	Операционная система коммутатора	Команда
Режим коммутатора и состояние канала	Система COS	`show fabric channel switchmode [mod]`
	Система IOS	`show fabric status [module {mod | all}]` `show fabric switching-mode [module {mod | all}]`
Разрешенные режимы коммутатора	Система COS	`show system switchmode`
	Система IOS	Нет
Счетчики канала коммутирующей матрицы	Система COS	`show fabric channel counters [mod]`
	Система IOS	`show fabric errors [module {mod | all}]`
Использование канала коммутирующей матрицы	Система COS	`show fabric channel utilization`
	Система IOS	`show fabric utilization [module {mod | all}]`

B.4: модуль FlexWAN коммутатора Catalyst 6000

- FlexWAN-модуль поддерживает до двух адаптеров WAN-портов маршрутизаторов Cisco серий 7200/7500 в любых комбинациях.
- В шасси коммутатора Catalyst 6000 можно установить до восьми FlexWAN-модулей.
- Могут использоваться любые адаптеры портов, кроме Ethernet, Fast Ethernet, Token Ring, FDDI, адаптеров порта канала (channel port adapters), модулей службы шифрования (encryption service modules), модулей службы сжатия (compression service modules) и адаптеров портов с двойной шириной (double-wide port adapters).

Конфигурация модуля

Конфигурирование интерфейсов на адаптерах FlexWAN-портов соответствует синтаксису команд программного обеспечения Cisco IOS. Конфигурационные команды вводятся либо в модуль MSFC/MSFC2, либо в модуль Supervisor коммутатора Catalyst 6000, использующий программное обеспечение собственной IOS. Прежде чем настраивать и использовать FlexWAN-модуль, в Supervisor необходимо установить как MSFC-, так и PFC-модуль.

При конфигурировании FlexWAN-интерфейсов необходимо использовать стандартный формат нумерации интерфейсов

Система COS	Нет
Система IOS	`interface` *mod/bay/port* (режим глобальной конфигурации)

Значение поля *mod* представляет собой номер модуля или гнезда FlexWAN-модуля. Поле *bay* определяет номер адаптера порта внутри FlexWAN-модуля. Значение поля *port* представляет собой номер порта или интерфейса внутри адаптера порта.

Полное описание команд Cisco IOS для конфигурирования интерфейсов приведено в книге David Hucaby and Steve McQuerry, *Cisco Field Manual: Router Configuration*, Cisco Press.

[redacted]

Приложение

Расширение VLAN-сетей в коммутаторах третьего уровня

Некоторые коммутаторы третьего уровня поддерживают функции третьего уровня только посредством аппаратного обеспечения интерфейса, и такие понятия, как VLAN-сети и базы данных VLAN-сетей, при работе с этими устройствами неприменимы. В категорию таких устройств также входят маршрутизаторы. В этом приложении описывается настройка широковещательного домена между портами на устройстве, имеющем только интерфейсы третьего уровня. В приложении также указано, каким образом следует связывать широковещательные домены с магистральными каналами ISL и 802.1Q для того, чтобы расширить VLAN-сеть коммутатора в устройстве третьего уровня.

- Для обработки пакетов на основании информации заголовков третьего уровня определено устройство, функционирующее только на третьем уровне, такое, как маршрутизатор или коммутатор 2948G-L3.
- В устройстве третьего уровня может быть активизировано программное обеспечение, поддерживающее функции моста, и порты могут назначаться мостовым группам.
- Все порты в общей мостовой группе находятся в одном широковещательном домене.
- Стандартно устройство третьего уровня пытается маршрутизировать IP-, IPX или AppleTalk-пакеты, если не задана иная конфигурация.
- Устройства третьего уровня не имеют сведений о VLAN-сетях, существующих в коммутаторах, но могут распознавать поступающие от коммутатора маркеры (tag) 802.1Q и ISL.
- При помощи связывания мостовой группы с VLAN сетью ISL или 802.1Q в устройстве третьего уровня можно расширить широковещательный домен виртуальной локальной сети

Расширение VLAN-сети с помощью интегрированной маршрутизации и функция мостового соединения (функция IRB)

Ниже описан процесс создания широковещательного домена внутри устройства третьего уровня и последующее его связывание с VLAN-сетью посредством магистрального канала.

1. Создание мостовой группы.

Система IOS	`bridge` *number* `protocol ieee` (режим глобальной конфигурации)

В режиме глобальной конфигурации команда `bridge` создает мостовую группу. Параметр *number* определяет эту группу, его значение может находиться в диапазоне от 1 до 255. Если VLAN-сеть на коммутаторе попадает в этот диапазон, рекомендуется создавать мостовую группу с тем же номером, что и VLAN-сеть. Значение параметра `protocol` при работе с коммутаторами всегда следует задавать равным `ieee`, поскольку коммутатор неспособен поддерживать *протокол распределенного связующего дерева (Spanning-Tree Protocol — STP)* версии DEC (Digital Equipment Corporation), и в сеть могут возникнуть мостовые петли.

2. Включение маршрутизации и мостовой обработки протокола.

Система IOS	`bridge irb` (режим глобальной конфигурации)

В режиме глобальной конфигурации команда `bridge irb` разрешает использование маршрутизации и мостовой обработки для конфигурируемых протоколов. По умолчанию коммутатор третьего уровня или маршрутизатор не поддерживают функции моста маршрутизируемых протоколов, таких, как IP, даже если интерфейс сконфигурирован для выполнения этих функций. Чтобы интерфейс выполнял функции моста для трафика, можно полностью отключить IP-маршрутизацию (маловероятно) или включить функцию IRB.

Внимание!

В зависимости от используемой операционной системы (OS) может возникнуть необходимость добавить другую команду, когда функция IRB активизирована. Команда `no bridge` *number* `route ip` может появиться в конфигурации после включения функции IRB. Эта команда отключает IP-маршрутизацию на любом интерфейсе, входящем в мостовую группу, заданную с помощью параметра *number*. Если команда не была введена автоматически, то для обеспечения поддержки функций моста между интерфейсами ее следует ввести вручную.

3. Назначение интерфейсов мостовой группе.

Система IOS	`bridge-group` *number* (режим конфигурирования интерфейса)

Порт добавляется к мостовой группе с помощью команды `bridge-group` в режиме конфигурирования интерфейса. Если порт является членом мостовой

группы, то все интерфейсы группы могут обмениваться данными на втором уровне. В сущности, интерфейсы находятся в одном и том же широковещательном домене, так же, как интерфейсы в одной VLAN-сети на коммутаторе второго уровня. Параметр *number* указывает, к какой группе принадлежит интерфейс. Когда интерфейс становится членом группы, на нем также запускается протокол STP.

4. Создание подынтерфейса на магистральном канале.

Система IOS	`interface` *type number.subinterfacenumber* (режим конфигурирования интерфейса)

Чтобы расширить VLAN-сеть из коммутируемой сети в маршрутизаторе или устройстве третьего уровня, необходимо, чтобы в трафике, поступающем от коммутатора второго уровня, было некоторое указание на VLAN-сеть, связанную с этим трафиком. Этого можно добиться путем использования магистрального канала (trunk link). На интерфейсе, подключенном к магистральному каналу коммутатора, создается подынтерфейс. Для создания подынтерфейса применяется команда `interface` с последующим указанием типа (*type* — Fast Ethernet или Gigabit Ethernet) и параметра *number.subinterfacenumber*. Каждая VLAN-сеть, которая будет связана с этим устройством третьего уровня, получит свой собственный подынтерфейс. Рекомендуется, чтобы номер (число, заданное после точки) совпадал с номером VLAN-сети, связанной с этим подынтерфейсом. Например, сеть VLAN 3, подключенная к интерфейсу FA 0/1, получила бы подынтерфейс FA 0/1.3.

5. Указание типа инкапсуляции и номера VLAN-сети.

Система IOS	`encapsulation {dot1q | isl}` *vlannumber* `[native]` (режим конфигурирования подынтерфейса)

На подынтерфейсе, созданном на этапе 4, необходимо указать инкапсуляцию магистрального канала и номер VLAN-сети, связанной с этим подынтерфейсом. Параметры указываются с помощью команды `encapsulation`. Тип магистрального канала задается с помощью команды `dot1q` и `isl`, а VLAN-сеть определяется параметром *vlannumber*.

Внимание!

В магистральных каналах стандарта 802.1Q маркеры для собственной VLAN-сети не устанавливаются. На коммутаторе третьего уровня необходимо указать параметр `native` для подынтерфейса, подключенного к собственной VLAN-сети 802.1Q, что позволит предотвратить маскирование сети (как правило, сети VLAN 1).

6. Добавление VLAN-сети к мостовой группе.

Система IOS	`bridge-group` *number* (режим конфигурирования подынтерфейса)

Путем добавления мостовой группы к подынтерфейсу из коммутатора к данной мостовой группе на маршрутизаторе или устройстве третьего уровня была добавлена VLAN-сеть. Это означает, что все интерфейсы на устройстве третьего уровня в данной мостовой группе, а также все порты на коммутаторе в данной

VLAN-сети находятся в одном широковещательном домене и фактически в одной VLAN-сети.

7. Маршрутизация для немаршрутизируемого трафика

а) Включение маршрутизации для мостовой группы

Система IOS	`bridge` *groupnumber* `route ip` (режим глобальной конфигурации)

IP-маршрутизация для интерфейсов в мостовой группе на устройстве третьего уровня в ходе этапа 2 была отключена вручную или автоматически маршрутизатором. Если требуется, чтобы устройство третьего уровня осуществляло функции маршрутизации для всех устройств в широковещательном домене (включая устройства во VLAN-сети коммутатора), то в первую очередь с помощью команды `bridge` *groupnumber* `route ip` необходимо включить маршрутизацию для данной мостовой группы.

б) Создание виртуального интерфейса третьего уровня

Система IOS	`interface bvi` *groupnumber* (режим глобальной конфигурации)

Если для интерфейсов была включена маршрутизация, то выполнение функций моста будет приостановлено до тех пор, пока не будет создан виртуальный интерфейс третьего уровня, предназначенный для использования этой мостовой группой. Виртуальный интерфейс для всех членов мостовой группы создается с помощью команды `interface bvi`.

в) Назначение IP-адреса интерфейсу третьего уровня

Система IOS	`ip address` *address mask* (режим конфигурирования интерфейса)

После создания интерфейса необходимо назначить ему IP-адрес. Этот адрес станет шлюзом для всех устройств в мостовой группе и VLAN-сети.

Пример конфигурирования функции

В приведенном ниже примере показаны сети VLAN 20, VLAN 55 и VLAN 103, связанные с мостовыми группами внутри коммутатора третьего уровня (такого, как 2948G-L3). Интерфейсы с 1 по 4 устройства третьего уровня необходимо связать с сетью VLAN 20, интерфейсы 11 и 12 — с сетью VLAN 55, а интерфейсы 25 и 26 — с сетью VLAN 103. Интерфейс G.49 обеспечит магистральное 802.1Q-соединение с коммутатором второго уровня. Коммутатор также будет предоставлять функции третьего уровня для VLAN-сетей 55 и 103. На рис. Г.1 показаны соединения для этого примера. Приведенную далее конфигурацию необходимо ввести в коммутатор третьего уровня.

```
L3Switch(config)# bridge 20 protocol ieee
L3Switch(config)# bridge 55 protocol ieee
L3Switch(config)# bridge 103 protocol ieee
L3Switch(config)# bridge irb
L3Switch(config)# interface FastEthernet 1
L3Switch(interface)# bridge-group 20
```

```
L3Switch(interface)# interface Fastethernet 2
L3Switch(interface)# bridge-group 20
L3Switch(interface)# interface Fastethernet 3
L3Switch(interface)# bridge-group 20
L3Switch(interface)# interface Fastethernet 4
L3Switch(interface)#bridge group 20
L3Switch(interface)#interface Fastethernet 11
L3Switch(interface)#bridge-group 55
L3Switch(interface)#interface Fastethernet 12
L3Switch(interface)#bridge-group 55
L3Switch(interface)#interface Fastethernet 25
L3Switch(interface)#bridge-group 103
L3Switch(interface)#interface Fastethernet 26
L3Switch(interface)#bridge-group 103
L3Switch(interface)#interface Gigabitethernet 49.20
L3Switch(interface-subif)#encapsulation dot1q 20
L3Switch(interface-subif)#bridge-group 20
L3Switch(interface-subif)#interface Gigabitethernet 49.55
L3Switch(interface-subif)#encapsulation dot1q 55
L3Switch(interface-subif)#bridge-group 55
L3Switch(interface-subif)#interface Gigabitethernet 49.103
L3Switch(interface-subif)#encapsulation dot1q 103
L3Switch(interface-subif)#bridge-group 103
L3Switch(interface-subif)#bridge 55 route ip
L3Switch(config)#bridge 103 route ip
L3Switch(config)#interface BVI 55
L3Switch(interface)#ip address 192.168.55.1 255.255.255.0
L3Switch(config)#interface BVI 103
L3Switch(interface)#ip address 192.168.103.1 255.255.255.0
```

Рис. 7.1. Расширение VLAN-сетей

Предметный указатель

M

H

O

P

С

Т

У

Ф

Ш

Научно-популярное издание

Дэвид Хьюкаби, Стив Мак-Квери

Руководство Cisco по конфигурированию коммутаторов Catalyst

Литературный редактор *И.В. Санин-Антонова*
Верстка *О.В. Линник*
Художественный редактор *В.Г. Павлютин*
Корректоры *З.В. Автократова, Л.А. Гордиенко, Л.В. Чернокозинская*

Издательский дом "Вильямс".
101509, Москва, ул. Лесная, д. 43, стр. 1.

Подписано в печать 14.09.2004. Формат 70×100/16.
Гарнитура Times. Печать офсетная.
Усл. печ. л. 43,15. Уч.-изд. л. 27,3.
Тираж 2500 экз. Заказ № 518.

Отпечатано с диапозитивов в ФГУП "Печатный двор"
Министерства РФ по делам печати,
телерадиовещания и средств массовых коммуникаций
197110, Санкт-Петербург, Чкаловский пр., 15

Руководство Cisco® по конфигурированию коммутаторов Catalyst®