УДК 004.056.53

Защита веб-ресурса от несанкционированного доступа с использованием метода аутентификации без необходимости ввода личных данных

Л.О. Воробьев, А.В. Чернышова
Донецкий национальный технический университет
lev.vorobjev@rambler.ru, chernyshova.alla@rambler.ru

Воробьев Л.О., Чернышова А.В. Защита веб-ресурса от несанкционированного доступа с использованием метода аутентификации без необходимости ввода личных данных. *В статье рассматривается метод аутентификации, позволяющий избавиться от необходимости ввода личных данных пользователя для доступа к веб-ресурсу. Проанализированы существующие методы аутентификации, определены их достоинства и недостатки. Описывается авторский алгоритм хеширования, приводится обоснование его применения в разработанном методе аутентификации.*

Ключевые слова*: аутентификация, авторизация, идентификация, несанкционированный доступ, информационная система*

Введение

Обеспечение защиты информации от несанкционированного доступа является актуальным вопросом в обеспечении компьютерной безопасности при разработке корпоративных информационных систем.

Постановка задачи

Цель: Разработка средств защиты информационного ресурса в Интернете от несанкционированного доступа (НСД) с авторизацией без необходимости ввода личных данных.

Перед разработчиками информационных систем часто возникает задача обеспечения защиты от несанкционированного доступа к разрабатываемому информационному ресурсу в Интернете. Предоставление доступа к защищаемому ресурсу для определенного пользователя называется авторизацией [1, с. 117], и включает процедуру идентификации пользователя и аутентификации для определения подлинности пользователя.

Необходим метод авторизации по идентификатору устройства и учетной записи в операционной системе. Этот метод аутентификации позволит освободить пользователя от необходимости ввода своих личных данных для регистрации в базе данных электронного ресурса и облегчить процедуру аутентификации.

В качестве информационного web-ресурса разрабатывается электронная зачетная система. Пользователи системы сохраняют результаты своей работы в виде заметок, отчетов и публикаций. Авторы регистрируются в организации с установленной системой, и необходимо обеспечить возможность работы с ресурсом без необходимости хранения их личных данных.

Для получения доступа к этому ресурсу необходимо иметь ключ, представленный цифрами и латинскими буквами. Один ключ выдается на одно устройство. При первом доступе к ресурсу в базе данных сохраняется зашифрованный хеш этого ключа и зашифрованный идентификатор устройства, с которого был получен доступ. Доступ с другого устройства с помощью этого ключа после его активации становится невозможным.

При получении доступа в файлах cookies сохраняется сертификат доступа в зашифрованном виде с ключом, который зависит от идентификатора устройства. Доступ к ресурсу с помощью этого cookies-файла возможен только с данного устройства и только при входе в учетную запись операционной системы. Копирование cookies на другое устройство не даст возможности злоумышленнику получить доступ к защищенному ресурсу.

Администратор защищаемого ресурса должен иметь возможность создавать новые ключи для регистрации пользователей и блокировать отдельные активированные ключи при необходимости.

Структура программной системы

Защищаемая программная система представляет собой информационную систему для хранения и редактирования заметок для электронной зачетной системы. Заметки сохраняются в базе данных под управлением СУБД PostgreSQL в зашифрованном виде.

Доступ к базе данных осуществляется по HTTP протоколу через удаленный сервер. Это позволяет пользователям взаимодействовать с системой посредством Интернет сервиса, мобильного телефона или программой на ПК, для того, чтобы создавать и синхронизировать редактируемые заметки.

Разрабатываемая программная система состоит из веб-сервера с базой данных и клиентских приложений. Структура системы описана на языке UML [2] в виде диаграммы компонентов (рисунок 1).

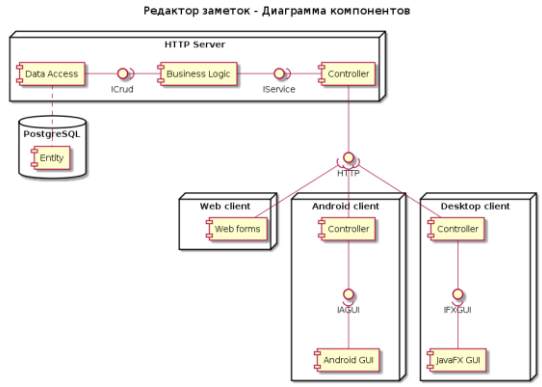

Рисунок 1 – Диаграмма компонентов разрабатываемой системы

Компонент DataAccess предназначен для доступа к базе данных PostgreSQL посредством объектно-реляционного отображения. Интерфейс CRUD предоставляет операции создания, чтения, обновления и удаления данных. Схема базы данных приведена на рисунке 2.

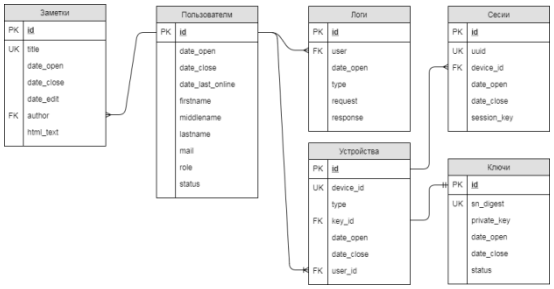

Рисунок 2 – Схема базы данных разрабатываемого ресурса

Таблица ключей содержит серийные номера в хешированном виде и секретные ключи ассиметричного алгоритма шифрования RSA для получения шифрованных данных от регистрируемого устройства.

Зарегистрированное устройство сохраняется в таблице устройств, в которой содержится идентификатор устройства для Android, или серийный номер жесткого диска для ноутбуков или ПК, или MAC-адрес сетевой карты, если авторизация проходит через веб-интерфейс. Тип идентификатора устройства определяется значением поля type в таблице устройств.

Для авторизации устройство должно предъявить серийный номер, хеш которого сравнивается со значением поля sn_digest ключа, зарегистрированного для данного устройства. При успешной авторизации генерируется UUID сессии, и передается в зашифрованном и подписанном виде клиенту. Ключ шифрования хранится в поле session_key таблицы сессий.

При необходимости, пользователь авторизованного устройства может создать учетную запись автора заметок и привязать её к своим устройствам для синхронизации. Учетные данные хранятся в таблице пользователей.

Порядок регистрации нового пользователя в системе показан на рисунке 3.

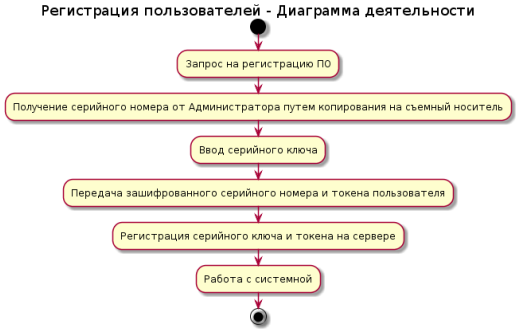

Рисунок 3 – Порядок регистрации пользователей в информационной системе

Серийный номер для регистрации нового устройства выдается администратором ресурса на съемном носителе вместе с открытым ключом шифрования. Конечный пользователь использует эти материалы для регистрации одного своего устройства, которое получит доступ к защищаемому устройству. Под токеном пользователя понимается зашифрованный идентификатор устройства. Сервер проверяет полученные зашифрованные данные, и если серийный номер действительно существует в базе данных и еще не был активирован, то новое устройство получает доступ к защищаемому ресурсу.

Стандартные алгоритмы аутентификации, достоинства и недостатки

Известные методы аутентификации:

– протокол аутентификации Нидхема-Шрёдера [3];

– протокол аутентификации по квитированию вызова [4, с. 860];

– аутентификация на основе одноразового пароля [4, с. 861];

– аутентификация на основе сертификатов [4, с. 863];

– протокол аутентификации OAuth 2.0 [5].

Протокол Нидхема – Шрёдера позволяет взаимно аутентифицировать две стороны посредством удостоверяющего центра. Реализацией данного протокола является система Kerberos [6], используемая в Microsoft ActiveDirectory и OpenLDAP для Linux. Сервер аутентификации хранит секретные ключи всех абонентов. Клиенты А и Б устанавливают защищенное соединение в следующей последовательности:

$$А \rightarrow СА: \{А, Б, I_{A1}\} \quad (1.1)$$

где I_{A1} – одноразовый идентификатор клиента А.

Получив это сообщение, СА извлекает из базы данных секретные ключи абонентов А и Б, вычисляет сессионный ключ К, и отправляет следующую информацию:

$$СА \rightarrow А: (I_{A1}, Б, К, \{К, А\}^{КБ})^{КА} \quad (1.2)$$

где КБ и КА – секретные ключи абонентов.

Субъект А расшифровывает полученный пакет, и передает субъекту Б данную информацию:

$$А \rightarrow Б: \{К, А\}^{КБ} \quad (1.3)$$

Субъект Б расшифровывает полученный пакет и отправляет А свой идентификатор:

$$Б \rightarrow А: \{ I_Б\}^{К} \quad (1.4)$$

Ожидается отклик абонента А:

$$А \rightarrow Б: \{I_Б - 1\}^{К} \quad (1.5)$$

Таким образом, оба абонента прошли взаимную аутентификацию.

Протокол Нидхема–Шрёдера обеспечивает надежную аутентификацию абонентов.

Аутентификация по квитированию вызова [7] заключается в передаче хешированного пароля между двумя абонентами. Схема аутентификации между клиентом А и сервером Б:

$$А \rightarrow Б: \{запрос\ на\ аутентификацию\} \quad (2.1)$$

$$Б \rightarrow А: \{ID, Challenge\} \quad (2.2)$$

$$А \rightarrow Б: \{ d(ID, Challenge, d(password)) \} \quad (2.3)$$

где d – функция хеширования.

Сервер сравнивает полученное значение с ожидаемым значением, и если результаты совпадают:

$$Б \rightarrow А: \{ Success \} \quad (2.4)$$

Надежность данного алгоритма аутентификации зависит от сложности пароля, поскольку известен случай взлома данного алгоритма с помощью алгоритма прямого перебора.

Для аутентификации по одноразовому паролю используется аппаратные или программные устройства, называемые токенами. Они вычисляют значение одноразового пароля по таймеру. Пользователь вводит этот пароль для аутентификации. Сервер выполняет вычисления по тому же алгоритму, что заложен в аппаратном ключе. Преимущество данного способа в надежности одноразового пароля по сравнению с условно-постоянным паролем.

При аутентификации на основе сертификатов информацию о пользователях предоставляют сами пользователи с помощью сертификатов, которые выдает централизованная организация. Сертифицирующие организации публикуют свои открытые ключи, необходимые для проверки цифровых сертификатов. Сертификат содержит: открытый ключ владельца сертификата; сведения о владельце; наименование сертифицирующей организации; электронная подпись сертификата, зашифрованная закрытым ключом организации. Данный метод используется, когда известно количество сертифицирующих организаций.

Метод аутентификации OAuth 2.0 позволяет получить доступ к нескольким сетевым ресурсам с помощью одной учетной записи без необходимости передачи логина и пароля. Уязвимость системы в том, что при получении несанкционированного доступа к одному ресурсу потенциальный злоумышленник получает доступ сразу ко всем ресурсам.

Также существует двухфакторная аутентификация, когда проверка осуществляется по нескольким аутентифицирующим особенностям: определенному свойству, знанию или владению. Такой подход обеспечивает большую надежность и используется в электронной почте или банковской системе.

Предлагаемый алгоритм аутентификации

Предлагаемый алгоритм аутентификации позволяет получить доступ к защищаемому ресурсу без необходимости передачи логина и пароля. Аутентификация основывается на владении секретным серийным номером для доступа к ресурсу с одного устройства. В отличии от стандартных способов авторизации, где регистрируются пользователи, предлагается способ регистрации устройств.

Авторизуемый пользователь обладает уникальным серийным номером SN и открытым ключом организации К для ассиметричного шифрования.

Клиент генерирует ключ сессии для защищенного соединения:

$$А \rightarrow Б: \{запрос\ на\ аутентификацию, SK\}^{K} \quad (3.1)$$

Сервер аутентификации отправляет зашифрованный ключ для аутентификации EK:

$$Б \rightarrow А: \{ID, Challenge, EK\}^{SK} \quad (3.2)$$

Клиент отправляет аутентифицирующие данные серверу:

А → Б: { ID, Challenge, HARD, d(SN)}^{EK}(3.3)
где HARD – серифный номер жесткого диска клиента.

Сервер аутентификации расшифровывает полученный пакет своим закрытым ключом, и если дайджест серифного номера по базе данных соответствует данному устройству, то возвращает сертификат доступа, зашифрованный секретным ключом СК:

Б → А: { PK, { UUID, HARD }^{CK} }^{EK}(3.4)
где UUID – уникальный идентификатор сессии; СК – сгенерированный секретный ключ данной сессии; РК – открытый ключ клиента для шифрования передаваемых данных повторной аутентификации.

Далее клиент сохраняет ключ РК и аутентифицируется с помощью полученного сертификата:

А → Б: {HARD, {UUID, HARD}^{CK} }^{PK}(3.5)

Сервер расшифровывает данный пакет и находит идентификатор сессии в базе данных. Если переданный идентификатор устройства совпадает с зашифрованным в сертификате и в базе данных, то клиенту предоставляется доступ.

В отличие от протокола Нидхема – Шрёдера, в предложенном способе уязвимыми являются серийные номера. Поэтому их необходимо хранить и передавать надежными каналами, через съемные носители.

Серийные номера генерируются на сервере, как случайные последовательности букв и цифр. В таблицу ключей записываются значения функции хеширования для сгенерированных ключей.

Для получения дайджеста предполагается использовать алгоритм MD5 [8]. Однако разрабатывается новый алгоритм, основанный на логической функции, описанной в следующем пункте.

Авторский алгоритм хеширования

Алгоритм вычисления дайджеста сообщения можно модифицировать, если использовать новый вид преобразования, изображенного на рисунке 4.

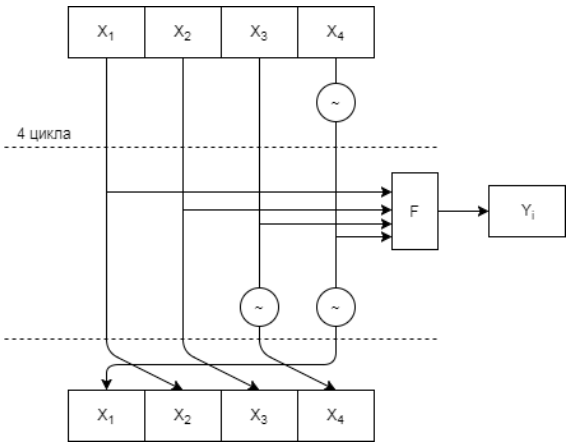

Рисунок 4 – Преобразование для новой функции хеширования

Побитовая функция F имеет следующий вид:

$F(X_1, X_2, X_3, X_4) = X_1 \wedge X_3 \vee X_2 \wedge X_4 \vee X_2 \wedge X_3$ (4.1)

где операции конъюнкции и дизъюнкции выполняются для каждого бита исходного значения.

Набор выходных значений определяется выражениями:

$Y_1 = F(X_1, X_2, X_3, \neg X_4)$; $Y_2 = F(X_4, X_1, X_2, \neg X_3)$; $Y_3 = F(X_3, X_4, X_1, \neg X_2)$; $Y_4 = F(X_2, X_3, X_4, \neg X_1)$; (4.2)

Таблица истинности функции (4.1) имеет одинаковое количество значений 1 и 0. Это гарантирует однозначность преобразования. Посредством использования основных законов булевой алгебры [9, с. 81] или с помощью пакета Wolfram Mathematica было несложно показать, что обратное преобразование возможно:

$X_1 = F(Y_1, Y_2, Y_3, \neg Y_4)$; $X_2 = F(Y_4, Y_1, Y_2, \neg Y_3)$; $X_3 = F(Y_3, Y_4, Y_1, \neg Y_2)$; $X_4 = F(Y_2, Y_3, Y_4, \neg Y_1)$; (4.3)

Поэтому для получения необратимого результата необходимо выполнить функцию свертки:

$S(Y_1, Y_2, Y_3, Y_4) = F(Y_1 >> 2, Y_2 >> 4, Y_3 >> 6, \neg Y_4)$ XOR $F(Y_4 >> 1, Y_1 >> 3, Y_2 >> 5, \neg Y_3 >> 7)$ (4.4)

где оператор >> обозначает побитовый циклический сдвиг.

Полученное в результате значение является функцией от 8 независимых аргументов и для нахождения исходного значения необходим полный перебор.

Выводы

Предложенный алгоритм решает проблему аутентификации без необходимости ввода личных данных. Традиционная авторизация в Интернете предполагает регистрацию пользователей с вводом имени пользователя и других его персональных данных. Обязательным пунктом регистрации является согласие пользователя на обработку

персональных данных. Предложенный метод аутентификации решает проблему регистрации, когда пользователь не согласен с последним пунктом.

Среди недостатков метода является слабая защищенность серверных ключей. Полученный ключ может быть передан третьим лицам до активации, что может привести к утечке информации. Поэтому предусмотрена возможность своевременного блокирования активированных ключей.

Предложена теоретическая возможность разработки нового алгоритма хеширования. Использование разработанного алгоритма повысит надежность предложенного метода аутентификации.

Литература

1. А.Ю. Щеглов «Защита компьютерной информации от несанкционированного доступа» [Текст] / А.Ю. Щеглов, – СПб.: Наука и техника, 2004 г. – 384 стр.

2. Г. Буч «Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ. Мухин Н.» [Текст] / Буч Г., Рамбо Д., Якобсон И., – М.: ДМК Пресс, 2006. – 496 с.: ил.

3. Ю.А. Семенов «Протокол аутентификации Нидхэма-Шредера в случаях симметричной и асимметричной системы шифрования» [Электронный ресурс], 2004. – Режим доступа: http://book.itep.ru/6/n_s_p_k.htm (дата обращения: 03.11.17)

4. В.Г. Олифер «Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 4-е изд.» / В.Г. Олифер, Н.А. Олифер – СПб.: Питер, 2010. – 2010. – 944 с.: ил.

5. RFC 6749 «Аутентификация OAuth 2.0»

6. RFC 1510 «Сервис аутентификации Kerberos 5.0»

7. RFC 2617 «Аутентификация по HTTP»

8. RFC 1321 «Алгоритм MD5»

9. И.А. Назарова «Дискретный анализ: учебно-методическое пособие» / И.А. Назарова – Донецк: ГВУЗ «ДонНТУ», 2012. – 277 с.

Воробьев Л.О., Чернышова А.В. Защита веб-ресурса от несанкционированного доступа с использованием метода аутентификации без необходимости ввода личных данных. *В статье рассматривается метод аутентификации, позволяющий избавиться от необходимости ввода личных данных пользователя для доступа к веб-ресурсу. Проанализированы существующие методы аутентификации, определены их достоинства и недостатки. Описывается авторский алгоритм хеширования, приводится обоснование его применения в разработанном методе аутентификации.*

Ключевые слова*: аутентификация, авторизация, идентификация, несанкционированный доступ, информационная система*

Vorobuev L.O., Chernyshova A.V. Protecting a web resource from unauthorized access using an authentication method without the need to enter personal data. *The article describes an authentication method that allows you to get rid of the need to enter the user's personal information to access the web resource. Existing methods of authentication are analyzed, their advantages and disadvantages are determined. The author's algorithm of hashing is described; the substantiation of its application in the developed method of authentication is given.*

Keywords*: authentication, authorization, identification, unauthorized access, information system*

Статья поступила в редакцию 20.03.2018
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом