§ КОДИРОВАНИЕ И ЗАЩИТА ИНФОРМАЦИИ

Войтюк Т.Е., Зудилова Т.В., Цымжитов Г.Б.

ЗАЩИТА ОТ ПОДБОРА ПАРОЛЕЙ ПРИ ДВУХФАКТОРНОЙ АУТЕНТИФИКАЦИИ

Аннотация: *Двухфакторная аутентификация требуется для установления безопасного соединения, когда удаленный пользователь пытается подключиться к корпоративным веб-сервисам. Аутентификации – это обязательное условие для веб-сервисов, которые обрабатывают конфиденциальную информацию. Двухфакторная аутентификация является способом улучшения корпоративной информационной безопасности. Существует множество готовых решений для реализации систем двухфакторной аутентификации, но эти решения имеют ряд недостатков, таких как: высокая стоимость или сложная интеграция в существующую корпоративную информационную структуру. Целью данного исследования является определение архитектуры системы, которая лишена перечисленных недостатков. Для проектирование архитектуры системы защиты от подбора паролей, предварительно был использован метод статического анализ данных для обоснования востребованности систем такого типа; метод анализа данных для определения требований к системам двухфакторной аутентификации; эксперимент подтвердил результаты исследования. Представленная архитектура обеспечивает защиту от подбора паролей, не зависит от дополнительных аппаратных или программных средств, имеет модульную структуру, которая дает преимущество в масштабировании. Архитектура определяет расширенный функционал для подобных систем: определение географического местоположения реальных IP-адресов, фильтрация адресов на основе геолокации и адреса прокси-сервера, используя запросы по методу POST, а также дает возможность формировать модули, которые легко интегрируется с существующую инфраструктуру предприятия. Результат применения предлагаемой системы показывает, что процент злоумышленников, обращающихся к корпоративной информационной системе, уменьшается.*

Ключевые слова: *двухфакторная аутентификация, аутентификационный код, одноразовый пароль, контроль прав доступа, сервис-ориентированная архитектура, технология*

DOI: 10.7256/2305-6061.2016.2.19025

единого входа, подбор паролей, информационная безопасность, безопасное соединение, веб-сервис

Abstract: *Two-factor authentication is required to establish a secure connection when a remote user tries to connect to the corporate web services. Authentication is a prerequisite for web services that process confidential information. Two-factor authentication is a way to improve the corporate information security. There are many ready solutions for the implementation of two-factor authentication system but these solutions have several disadvantages, such as high cost or difficult integration into existing corporate information structure. The aim of this study is to define the architecture of the system that overcomes the mentioned disadvantages. For designing a protection system against password guessing the authors previously used a method of static analysis to justify the demand for systems of this type. The authors also used data analysis method to determine the requirements for the system of two-factor authentication; an experiment confirmed the results of a research. Presented architecture provides protection from password guessing, does not depend on additional hardware or software and has a modular structure, which gives the advantage of scalability. The architecture defines advanced functionality for such systems: determining geographic location of real IP-addresses, address filtering based on geolocation and proxy addresses using a POST requests. It also allows building modules, which can be easily integrated with existing enterprise infrastructure. The result of using the proposed system shows that the percentage of intruders accessing corporate information system is reduced.*

Keywords: *information security, password guessing, single sign-on technology, service-oriented architecture, control permissions, one-time password, authentication code, two-factor authentication, secure connection, web service*

В настоящее время проблема информационной безопасности в корпоративных информационных системах очень остро стоит перед компаниями любого уровня. В последнее время стали частым явлением заказные атаки на информационные ресурсы компаний, утечка критически важной корпоративной информации, рост объемов паразитного трафика, вымогательство и шантаж. Поэтому исследования в области обеспечения безопасности корпоративных информационных систем (ИС), все еще остаются актуальными, несмотря на большое количество уже проведенных исследований. Результатами исследований в данной области становятся различные подходы к организации безопасности корпоративных информационных систем[1] или новые методы/алгоритмы, позволяющие исправлять уязвимости в безопасности[2]. Однако до настоящего времени не создано полностью безопасной корпоративной информационной системы, так как самым слабым звеном в информационной безопасности является пользователь, и это подтверждают слова Брюса Шнейера, о том, что безопасность является сильной лишь тогда, когда существует слабое звено и этим слабым звеном является пользователь[3].

Пользователи получают доступ к важной информации компании, используя аутентификацию в корпоративной информационной системе. Наиболее распространенным механизмом аутентификации пользователей в таких системах является использование

DOI: 10.7256/2305-6061.2016.2.19025

текстовых паролей. Для предотвращения несанкционированного доступа необходимо, чтобы пользователь выбирал сложные надежные пароли и хранил их согласно внутренней политике безопасности компании. Однако не все пользователи готовы соблюдать внутренние политики. Они могут менять пароли на более простые, записывать их в личный кабинет публичного облака или записную книжку, использовать один и тот же пароль для доступа ко всем ресурсам, требующим аутентификации. Отследить данное поведение пользователей очень сложно, поэтому возрастает вероятность утечки аутентификационных данных. В настоящее время доступ злоумышленников к корпоративным сетям за счет похищения аутентификационных данных пользователей стало настоящей эпидемией.

На рисунке 1 представлено распределение интернет-трафика на защищаемый неиндексируемый корпоративный ресурс, использующий UserAgent 2013, 2014 и 2015 годы.

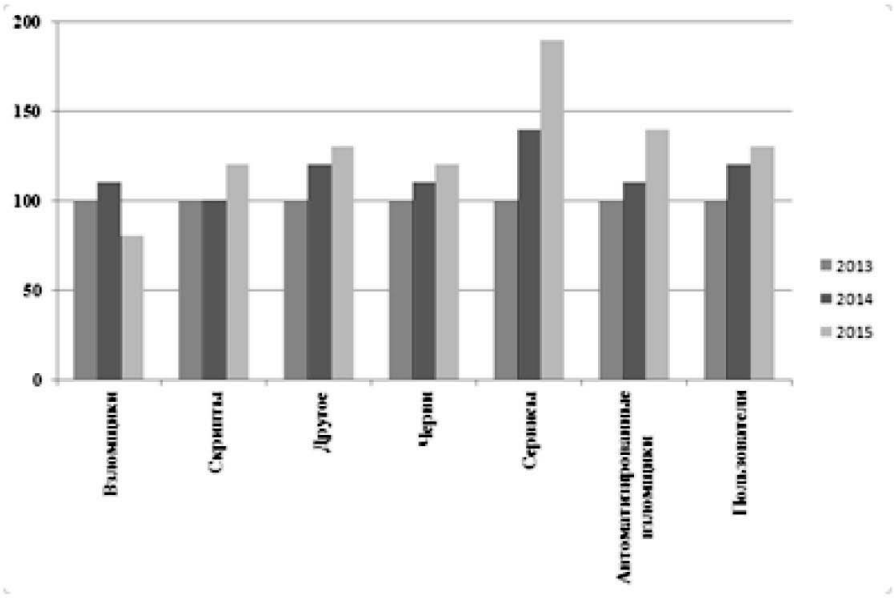

Рисунок 1 - Распределение интернет-трафика в корпоративной инфраструктуре

Рисунок 1 показывает, что с каждым годом увеличивается количество пользователей корпоративного ресурса и количество сервисов, предоставляемых этим ресурсом. При этом уменьшается количество вредоносных скриптов, которые создаются низкоквалифицированными злоумышленниками. Однако резко возрастает количество автоматизированных атак и не снижается количество целенаправленных атак. Эти данные позволяют сделать вывод, что количество высококвалифицированных злоумышленников возрастает.

Чтобы уменьшить количество несанкционированных обращений в информационных системах внедряют метод 2-факторной аутентификации.

Двухфакторная аутентификация требует от пользователя два типа паролей для доступа к корпоративной системе.

Обычно в корпоративных системах первым типом аутентификации является текстовый пароль.

DOI: 10.7256/2305-6061.2016.2.19025

Вторым типом аутентификации является одноразовый пароль (англ. One Time Password, сокр. OTP), который отправляется пользователю или синхронизируется в зависимости от алгоритма, метода аутентификации.

Преимущества двухфакторной аутентификации очевидны:

- повышение надежности аутентификации, которая также позволяет бороться с атаками посредника (англ. Man in the middle, сокр. MitM) [4] и подбором паролей;
- повышение вероятности того, что действие было выполнено действительно пользователем, предъявившим идентификатор, т.е. обеспечение неотказуемости пользователя.
- упрощение работы с информационной системой пользователям, т.к. не требуется запоминать множество паролей, не требуется производить периодическое изменение паролей и т.д.

Но наряду с преимуществами существует также и ряд недостатков:

- более дорогостоящее решение, зачастую зависящее от архитектуры системы;
- токены, карточки и другие средства 2х-факторной аутентификации могут быть утеряны, оставлены дома (или в других местах) или повреждены. Без помощи системного администратора работа таких пользователей останавливается;
- не все приложения поддерживают 2х-факторную аутентификацию, т.е. полный отказ от паролей невозможен.

Благодаря внедрению корпоративных систем с технологией единого входа (англ. Enterprise Single Sign-On, сокр. ESSO), которые являются фактически прослойкой между двухфакторной аутентификацией и различными приложениями, два последних недостатка становятся несущественными. Благодаря тому, что ESSO-система обладает широким функционалом, первоначальная авторизация пользователя производится двухфакторным методом именно в ESSO, а затем ESSO взаимодействует с прикладными системами, выбирая из своей базы соответствующие данному пользователю и конкретной системе учетные данные. В ESSO происходит управление всеми ключевыми носителями, привязка учетных данных, генерация паролей, блокировка, назначение ролей и т.д. Однако внедрение ESSO приводит к усложнению архитектуры корпоративной информационной системы и увеличивает расходы на внедрение и эксплуатацию системы аутентификации. Для компаний среднего уровня стоимость такого решения является неприемлемой, поэтому зачастую средний и малый бизнес отказывается от двухфакторной аутентификации.

В данном исследовании представлен подход к организации архитектуры корпоративной информационной системы, поддерживающей двухфакторную авторизацию без привлечения дополнительных затрат на установку и обслуживание ESSO-систем. Основным требованием к архитектуре является модульность и многоступенчатая система фильтрации, поскольку злоумышленник может нанести вред компании, не только получив аутентификационные данные, но и создав очередь в стеке отправки одноразовых паролей для потенциальных пользователей системы, увеличив тем самым расходы на отправку, например при СМС аутентификации.

В данной части рассматриваются алгоритмы получения одноразовых паролей. Одноразовый пароль – это пароль, используемый только для одного сеанса аутентификации.

DOI: 10.7256/2305-6061.2016.2.19025

При генерации OTP используют случайные числа. Конкретные алгоритмы получения случайного числа сильно различаются, однако их все можно разделить на две группы:

1. Алгоритмы, позволяющие математически выводить псевдослучайное число. В свою очередь делятся на алгоритмы на основе предыдущих паролей, где важен порядок паролей, и на основе запроса, где случайное число выбирается сервером и/или формируется счетчиком[5].

2. Алгоритмы, основанные на синхронизации по времени. В этих алгоритмах пароль действует в течение короткого периода времени.

По способу доставки OTP можно классифицировать на:

- текстовые отправления(CMC, e-mail);
- программные токены, формируемые мобильными приложениями(Google Auth);
- аппаратные токены (RSA SecurID);
- программные токены, формируемые веб-приложениями;
- напечатанные одноразовые пароли на бумаге или скретч-карте, которую при аутентификации необходимо иметь при себе.

Наиболее распространенными алгоритмами, позволяющими получать одноразовые пароли, являются: OCRA, HOTP и TOTP.

HOTP (англ. HMAC-Based One-Time Password Algorithm) – это алгоритм, базирующийся на алгоритме HMAC (англ. hash-based message authentication code), где HMAC является аутентификационным кодом и на основе хэш-кода сообщения. Данный алгоритм описан в стандарте RFC 4226[6]

Общая схема аутентификации клиента по алгоритму HOTP состоит из двух этапов.

Сначала вычисляется код аутентификации по алгоритму HMAC, затем функция усечения результата (Truncate) применяется для полученного кода [7].

Код аутентификации вычисляется по формуле 1:

$$HMAC(k,text) = H(K \text{ XOR } opad, H(K \text{ XOR } ipad, text)), \quad (1)$$

где H - криптографическая хэш-функция, K – ключ, ipad и opad – константы, магические числа [8], обычно равные 0x36 или 0x5C[9]. Переменная text - есть некоторый смещаемый фактор, например некий счетчик или известные сообщения.

Так как одноразовый пароль является вторым фактором при двухфакторной аутентификации, и он используется один раз при создании сеанса, то можно сделать вывод, что алгоритм HOTP является кодом HMAC по известной криптографической хэш-функции, известной обоим участником обмена данными (клиенту и серверу). Обычно в качестве криптографической функции используют хэш-функцию SHA-1.

После получения кода HMAC необходимо применить функцию усечения результата (2) для аутентификации по HOTP:

$$HOTP(K,text) = Truncate(HMAC\text{-}SHA1(K,text)). \quad (2)$$

Именно функция Truncate позволяет преобразовать значение HMAC-SHA-1 в значение HOTP [6].

TOTP (англ. Time-based One Time Password Algorithm) – это улучшенный алгоритм на

DOI: 10.7256/2305-6061.2016.2.19025

основе HOTP. Данный алгоритм описан в стандарте RFC 6238 [10]. Для формирования одноразового пароля в данном алгоритме используется время в формате Unix-time, в которое произошел запрос. Именно время является фактором, влияющим на изменение сообщения. Для выполнения алгоритма время конвертируется из формата даты в числовой формат и представляется в секундах с момента начало Эпохи Unix, за начало отсчета берется 1 января 1970 года.

Для функционирования алгоритма необходимо произвести синхронизацию времени между участниками обмена сообщениями, а также создать временное окно (например, 30 секунд), которое позволит получить результат в случае временных потерь, связанных с передачей сообщения по сети.

Для алгоритма TOTP пароль будет вычислять по формуле 3:

$$HMAC(K, unix\_timestamp / 30). \quad (3)$$

OCRA (англ. Challenge-Response Algorithm) – этот алгоритм, базирующийся на двух предыдущих, описан в стандарте RFC 6287[11]. Пароль, при использовании данного алгоритма, вычисляется по формуле 4:

$$OCRA = CryptoFunction(K, DataInput), \quad (4)$$

где CryptoFunction – это функция, выполняющая вычисления, используя секретный ключ K, известный обоим участникам, и структуру DataInput.

DataInput содержит несколько параметров, объединенных в одну строку (5):

$$Datainput = (OCRASuite \mid 00 \mid C \mid Q \mid P \mid S \mid T), \quad (5)$$

где 00 – используется в качестве разделителя; C – счетчик длиной в 8 байт, должен быть синхронизирован между клиентом и сервером; Q – запрос длиной в 128 байт, в случае меньшей длины его нужно дополнить нулями; P – это хэш-функция от PIN-кода, который знают клиент и сервер; S – строка до 512 байт, описывает состояние сессии, длина указана в OCRASuite; T – количество прошедших промежутков времени от условной точки отсчета, за которую принята дата 1 января 1970 года, длина 8 байт, единицы измерения указаны в OCRASuite; OCRASuite – это строка, содержащая набор параметров, для формирования ответа.

Для двусторонней аутентификации и подписи, клиент и сервер должны обменяться двумя строками OCRASuite: одна для сервера, другая для клиента.

Часто при проектировании системы двухфакторной аутентификации специалисты по безопасности создают собственные закрытые алгоритмы, которые в основном базируются на HOTP, но имеют некоторые отличия.

Например, для обеспечения дополнительной защиты от подбора пароля, можно скрыть алгоритм генерации модифицируемого параметра text, в этом случае код аутентификации будет формироваться по формуле 6:

$$OTP = Substring(105,SHA3\text{-}512(k, secretfunc(text))), \quad (6)$$

где secretfunc – функция получения кода аутентификации, используя собственный алгоритм.

DOI: 10.7256/2305-6061.2016.2.19025

Таким образом, из представленных алгоритмов видно, что они являются достаточно простыми в реализации и их чаще всего используют для получения второго фактора в системах двухфакторной аутентификации. Данные алгоритмы будут использоваться в предлагаемом архитектурном решении системы двухфакторной аутентификации.

При разработке архитектуры системы двухфакторной аутентификации были проанализированы основные технические требования, которые предъявляются к многофакторным системам аутентификации, основываясь на источниках [12 - 15]. Эти исследования помогли определить следующие минимальные требования к архитектуре ИС системы предприятия, поддерживающей двухфакторную аутентификацию:

- Динамическая авторизация по IP пользователей веб-сервисов компании;
- Возможность добавления аутентификации любой службы, например, SMTP, IMAP, Jabber, FTP и других, что делает более комфортной работу для конечных пользователей;
- Возможность разграничения доступа (фильтрации) для веб-служб: почта, трекер и другие;
- Авторизация новых адресов должна производиться путем ввода разового пароля, получаемого пользователем на заранее зарегистрированные мобильный телефон или электронную почту;
- Синхронизация с внешними базами данных для пользователей Active Directory или openLDAP, возможность определения полей атрибутов таблиц для синхронизации;
- Мобильные телефоны пользователей должны храниться в LDAP, портальной адресной книги или локальной базе данных. Необходимо реализовать, по крайней мере, один тип хранения;
- Система должна быть модульной, поскольку она дает возможность заменить: модули без серьезных последствий для системы; тип аутентификации; тип межсетевого экрана или другого программного обеспечения;
- Фильтрация пользователей по предопределенной группе (базовой группе). Базовая группа/роль дает пользователю доступ к основному набору сервисов;
- Возможность добавления дополнительных групп для отдельных сервисов и фильтрация по ним;
- Плавающий срок хранения авторизованных уникальных идентификаторов (англ. unique identifier, сокр. UID) сеанса, учитываться должны следующие параметры: давность последнего входа (таймаут), общее кол-во входов;
- Это зависит от следующих параметров: удаленность последнего входа (тайм-аута) и общего количества входа;
- Поддержка CAPTCHA и таймаута на повторную отправку, например, 5 минут после 2-3 отправок разовых паролей подряд, после 5-10 отправок — блокировка;
- Обязательное информирование пользователя при удачной и неудачной аутентификации. Алгоритм информирования должен быть следующий: если аутентификация прошла успешно, то на несколько секунд показывается баннер «Успешный

DOI: 10.7256/2305-6061.2016.2.19025

вход в систему», после чего идет перенаправление на искомый адрес; если аутентификация завершилась неудачей, то показывается сообщение с ошибкой, контакты технической поддержки и ссылка на повторную аутентификацию.

- Хранение всех ip-адресов с которых пользователь получал аутентификацию;
- Хранение всех ip-адресов с которых пользователь запрашивал аутентификацию;
- Ограничение по ip-адресу геолокаций;
- Универсальность по соотношению внешних/внутренних адресов;
- Доступность ряда сервисов только для доверенных групп IP (реализуется опциональным фильтром);
- Возможность ограничения серверов электронной почты, которые могут быть использованы для получения разового пароля.

В ходе проведенного исследования были также изучены готовые архитектурные решения ИС, поддерживающей двухфакторную аутентификацию. Рассматривались системы: SecurEnvoy, Yubico и Duo Security. Однако ни одно из этих решений не удовлетворяет требованиям к системам двухфакторной аутентификации, описанным выше. Это послужило причиной проектирования архитектуры системы защиты от подбора паролей при двухфакторной аутентификации.

Архитектура системы представлена на рисунке 2. Система имеет модульную структуру, что помогает интегрировать существующие решения двухфакторной аутентификации, системы фильтрации и систем обнаружения и предотвращения вторжений.

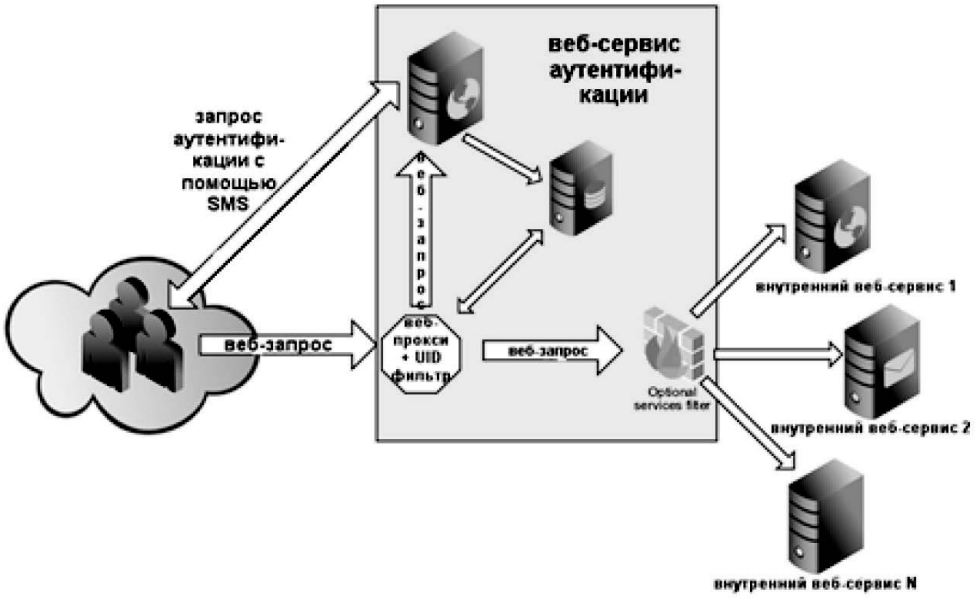

Рисунок 2 – Архитектура системы защиты от подбора паролей

Алгоритм функционирования системы состоит в следующем:

1. Клиент из Интернета запрашивает доступ к корпоративному веб-ресурсу.
2. Запрос оценивается веб-прокси с помощью белых и черных списков.

DOI: 10.7256/2305-6061.2016.2.19025

3. Параметры запроса проверяются на наличие существующих UID сессии. Если UID не находится в базе данных, и не существует в серых и черных списках, то запрос пересылается на веб-ресурс предприятия.
4. Начальной страницей веб-ресурсов предприятия является страница аутентификации. Клиент вводит ESSO логин и CAPTCHA.
5. Если первая аутентификация прошла успешно, пользователь должен ввести код подтверждения (второй фактор) в странице подтверждения. Второй фактор (OTP код) посылает через выбранный способ доставки (CMC, электронная почта) или генерируется с помощью Google Auth или скретч-карты.
6. Код подтверждения проверяется алгоритмами: HOTP, OCRA, TOTP или другими из внутренней базы данных.
7. Создается и добавляется UID, также IP добавляется в списки доверенных.
8. Происходит перенаправление к нужному корпоративному ресурсу.

Предлагаемая архитектура системы защиты от подбора пароля для двухфакторной аутентификации соответствуют предъявляемым требованиям к системам многофакторной аутентификации. Модульность архитектуры способствует удобству внедрения и управления этой системой, также позволяет обезопасить сразу все корпоративные веб-сервисы, то есть инфраструктуру предприятия.

На рисунке 3 легко увидеть улучшение защиты от целенаправленных атак (злоумышленников) при внедрения данной архитектуры.

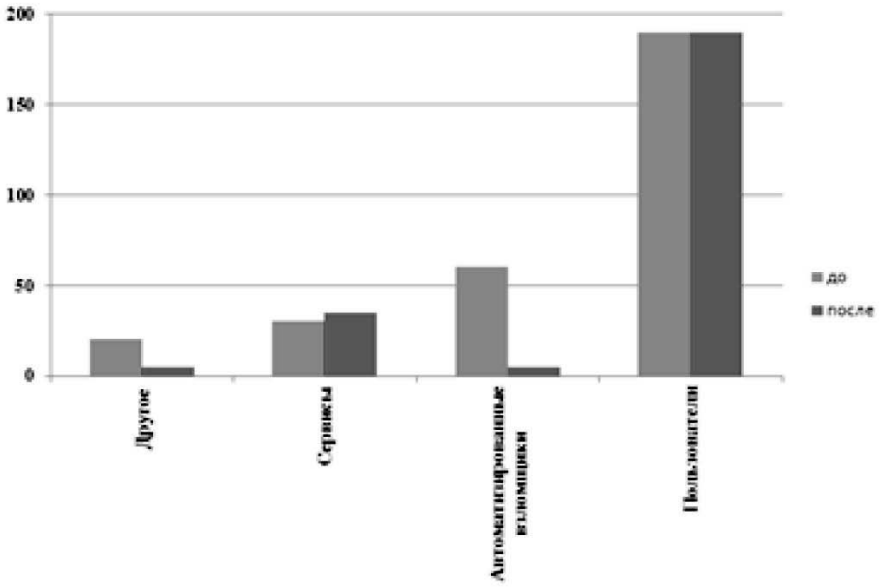

Рисунок 3 – Веб-трафик до и после применения системы

Из графика видно, что процент отражения негативного трафика увеличивается при его точечном разборе. Так после внедрения предложенной архитектуры количество автоматизированных взломщиков уменьшилось на 86%.

DOI: 10.7256/2305-6061.2016.2.19025

Дальнейшим улучшением предложенной архитектуры может являться применение алгоритмов машинного обучения, что сможет повысить процент отраженного негативного трафика при атаке на корпоративную ИС.

Библиография :

1. Pramanik S., Security Architecture Approaches. The Journal of Defense Software Engineering, 2013. 26(6): pp. 12-17.
2. Bürger J., Jürjens J., Wenzel S., Restoring security of evolving software models using graph transformation international. Journal on Software Tools for Technology Transfer, 2015, 17(3): pp. 267-289.
3. Киви Берд. Квантовая криптонеопределенность. Журнал «Компьютерра», 2004, №46. [Электронный ресурс]. – Режим доступа: http://old.computerra.ru/206396/ свободный. Яз. русс. (дата обращения 20.01.2016).
4. Nam S.Y., Djuraev S., Collaborative approach to mitigating ARP poisoning-based Man-in-the-Middle attacks. Computer Networks: The International Journal of Computer and Telecommunications Networking, 2013. 57(18): pp. 3866–3884.
5. Eastlake D., Crocker S., Schiller J. RFC 1750: Randomness Recommendations for Security. RFC Editor, 1994, p. 30.
6. M'Raihi D., Bellare M., Hoornaert F., Naccache D., Ranen O.. RFC 4226. HOTP: An HMAC-Based One-Time Password Algorithm. RFC Editor, 2005, p. 37.
7. Bellare M., Canettiy R., Krawczykz H. Message Authentication using Hash Functions The HMAC Construction. CryptoBytes. 1996. 2(1): pp.1-2.
8. Фергюсон Н., Шнайер Б. Практическая криптография. М: Вильямс, 2005, 424 с.
9. Бабаш А.В., Баранова Е.К. Криптографические методы защиты информации. Учебник. Серия: "Бакалавриат и магистратура". М: Кнорус, 2016, 189 с.
10. M'Raihi D., Machani S., Pei M., Rydell J. RFC 6238. TOTP: Time-Based One-Time Password Algorithm. RFC Editor, 2011, 16 p.
11. M'Raihi D., Rydell J., Bajaj S., Machani S., Naccache D. RFC 6287. OCRA: OATH Challenge-Response Algorithm. IETF, 201, p. 38.
12. Liu J., Liu C., Jiao D., Chen J. The Research of a Multi-Factor Dynamic Authorization Model. Proceedings of the 2012 IEEE Ninth International Conference on e-Business Engineering: 2012, pp. 201-205.
13. Oh S. W., Kim H. Decentralized access permission control using resource-oriented architecture for the web of things. Advanced Communication Technology (ICACT), 2014 16th International Conference: 2014, pp. 749-753.
14. Al-Kahtani M.A., Sandhu R.S. 2002. A Model for Attribute-Based User-Role Assignment. Proceedings of the 18th Annual Computer Security Applications Conference: 2002, pp. 353-362.
15. Razzaq A., Hur A., Shahbaz S., Masood M., Ahmad R., Critical analysis on web application firewall solutions. IEEE Eleventh International Symposium on Autonomous Decentralized Systems: 2013, pp. 1-6.

References:

1. Pramanik S., Security Architecture Approaches. The Journal of Defense Software Engineering, 2013. 26(6): pp. 12-17. 2.

DOI: 10.7256/2305-6061.2016.2.19025

2. Bürger J., Jürjens J., Wenzel S., Restoring security of evolving software models using graph transformation international. Journal on Software Tools for Technology Transfer, 2015, 17(3): pp. 267-289.
3. Kivi Berd. Kvantovaya kriptoneopredelennost'. Zhurnal «Komp'yuterra», 2004, №46. [Elektronnyi resurs]. – Rezhim dostupa: http://old.computerra.ru/206396/ svobodnyi. Yaz. russ. (data obrashcheniya 20.01.2016).
4. Nam S.Y., Djuraev S., Collaborative approach to mitigating ARP poisoning-based Man-in-the-Middle attacks. Computer Networks: The International Journal of Computer and Telecommunications Networking, 2013. 57(18): pp. 3866–3884.
5. Eastlake D., Crocker S., Schiller J. RFC 1750: Randomness Recommendations for Security. RFC Editor, 1994, p. 30.
6. M'Raihi D., Bellare M., Hoornaert F., Naccache D., Ranen O.. RFC 4226. HOTP: An HMAC-Based One-Time Password Algorithm. RFC Editor, 2005, p. 37.
7. Bellare M., Canettiy R., Krawczykz H. Message Authentication using Hash Functions The HMAC Construction. CryptoBytes. 1996. 2(1): pp.1-2.
8. Fergyuson N., Shnaier B. Prakticheskaya kriptografiya. M: Vil'yams, 2005, 424 c.
9. Babash A.V., Baranova E.K. Kriptograficheskie metody zashchity informatsii. Uchebnik. Seriya: "Bakalavriat i magistratura". M: Knorus, 2016, 189 c.
10. M'Raihi D., Machani S., Pei M., Rydell J. RFC 6238. TOTP: Time-Based One-Time Password Algorithm. RFC Editor, 2011, 16 p.
11. M'Raihi D., Rydell J., Bajaj S., Machani S., Naccache D. RFC 6287. OCRA: OATH Challenge-Response Algorithm. IETF, 201, p. 38.
12. Liu J., Liu C., Jiao D., Chen J. The Research of a Multi-Factor Dynamic Authorization Model. Proceedings of the 2012 IEEE Ninth International Conference on e-Business Engineering: 2012, pp. 201-205.
13. Oh S. W., Kim H. Decentralized access permission control using resource-oriented architecture for the web of things. Advanced Communication Technology (ICACT), 2014 16th International Conference: 2014, pp. 749-753.
14. Al-Kahtani M.A., Sandhu R.S. 2002. A Model for Attribute-Based User-Role Assignment. Proceedings of the 18th Annual Computer Security Applications Conference: 2002, pp. 353-362.
15. Razzaq A., Hur A., Shahbaz S., Masood M., Ahmad R., Critical analysis on web application firewall solutions. IEEE Eleventh International Symposium on Autonomous Decentralized Systems: 2013, pp. 1-6.

DOI: 10.7256/2305-6061.2016.2.19025