Подход к оцениванию устойчивости процесса функционирования Веб-браузеров с использованием серверов

К. В. Торгашов, email: torgashov18@mail.ru[1]
А. В. Алямкин, email: Alexandralyamkin@yandex.ru[1]
М. А. Кожевников, email: kmaxss@yandex.ru[1]
Г. В. Васильев, email: gleb_mesarthim@mail.ru[1]

[1] Краснодарское высшее военное училище им. С. М. Штеменко

Аннотация. *В данной работе проводится исследование кросс-протокольных атак на TLS в целом и систематическое исследование на примере веб-серверов, перенаправляющих HTTPS-запросы от веб-браузера жертвы на SMTP, IMAP, POP3 и FTP-серверы.*

Ключевые слова: *кросс-протокольная атака, веб-браузер, веб-сервер, сниффинг, SMTP, IMAP, POP3, FTP*

Введение

TLS широко используется для обеспечения конфиденциальности, подлинности и целостности протоколов прикладного уровня, таких как HTTP, SMTP, IMAP, POP3 и FTP [1-5]. Однако TLS не связывает TCP-соединение с целевым протоколом прикладного уровня. Это позволяет злоумышленнику перенаправить трафик TLS на другую конечную точку службы TLS на другом IP-адресе и/или порту [6].

Провести оценку реального ущерба от атаки веб-браузеров и широко распространенных серверов электронной почты и FTP можно в лабораторных экспериментах и при сканировании всего Интернета. Сделать это можно посчитав количество веб-серверов, уязвимых для кросс протокольных атак, узнав процент веб-серверов, которые могут быть атакованы с использованием уязвимого сервера приложений и рассмотрев эффективность расширений TLS, таких как Application Layer Protocol Negotiation (ALPN) и Server Name Indiciation (SNI), в смягчении этих и других межпротокольных атак.

1. Оценка веб-браузеров

Была произведена оценка поведения браузеров, относящееся к кросс-протокольным атакам, для Chrome 86, Firefox 81, Internet Explorer 11, Edge Legacy 44, Edge 86, Opera 71 и Safari 14 путем ручного доступа к пользовательскому веб-серверу с одной тестовой страницей для

© [1] Торгашов К. В., Алямкин А. В., Кожевников М. А., Васильев Г. В., 2021

каждого оцениваемого свойства. Результаты исследования показаны на рис. 1.

Results	Chrome	Firefox	IE	Edge Legacy	Edge	Opera	Safari
Header Lines	17	11	11	12	17	17	11
Content Sniffing	○	○	●	●	○	○	○
Keep-Alive							
w/ HTTP/1.1	●	●	●	●	●	●	●
w/ noise	○	○	○	○	○	○	○

● Support ○ No support

Рис. 1. Поведение браузера, относящееся к кросс-протокольным атакам

Оцениваемые параметры описанных выше браузеров:

Количество строк заголовка. Для каждого браузера было определено, сколько строк заголовка включено в типичный POST-запрос, отправляемый браузером. Это также минимальное количество ошибок, которое должен допускать сервер приложений с текстовым протоколом на основе строк, чтобы быть пригодным для кросс-протокольных атак. Было установлено, что все браузеры отправляют заголовки, состоящие из 11-17 строк.

Контент-сниффинг. Атаки отражения и загрузки могут быть чувствительны к шуму в протокольных данных, возвращаемых сервером приложений. Для каждого браузера было проверено, выполняет ли браузер сниффинг содержимого и выполняет ли он встроенный JavaScript в любом случае. Было подтверждено, что это действительно так для Internet Explorer и Edge Legacy, в то время как все остальные протестированные браузеры не выполняют контент-сниффинг и, следовательно, не выполняют JavaScript в шумных ответах.

Повторное использование соединения. Отправка более одного запроса в одном межпротокольном соединении может быть выгодна злоумышленнику. Стандарт HTTP/1.1 по умолчанию определяет постоянные соединения между клиентом и сервером для нескольких HTTP-запросов. Таким образом, браузер должен повторно использовать TCP-соединение, если используется стандарт HTTP не ниже версии 1.1 и сервер не отправил HTTP-заголовок Connection: close.

Наличие данного поведения было проверено для Chrome, Edge, Edge Legacy, Firefox, Internet Explorer, Opera и Safari. Все браузеры повторно используют соединение после получения действительного ответа, содержащего как минимум строку Status-Line и Content- Length. Chrome, Firefox и Opera даже принимают строку состояния, состоящую только из HTTP-Version без Status-Code или Reason-Phrase. Internet Explorer и Edge требуют полную Status-Line. Ни один из браузеров не использует соединение повторно, если первая строка ответа не начинается с маркера HTTP. Это актуально для атак на загрузку, основанных на повторном использовании соединения, которые включают протокольный шум в начале ответа, что сводит на нет цель отправки нескольких запросов в одном и том же TLS-соединении.

2. Стратегии нападения

Кросс-протокольные атаки на HTTPS могут быть выполнены с использованием SMTP, IMAP, POP3 и FTP в качестве серверов приложений. С помощью этих протоколов могут быть произведены атаки на загрузку, скачивание и отражение [7-12].

Были определены следующие стратегии для реализации атак загрузки, скачивания и отражения на HTTPS с использованием серверов приложений SMTP, IMAP, POP3 и FTP:

Атаки на отражение. Все описанные выше протоколы являются линейными протоколами. Они интерпретируют каждую строку запроса HTTPS как команду и генерируют ответ на каждую. Если реализация получает команду, она может использовать некоторые данные из ввода в своем ответе. Например, отправка команды HELP <script> attack();</script> на FTP-сервер может привести к ответной команде Unknown command:<script>attack();</script>. Обычно, как и в случае с SMTP, POP3 и FTP, наличие таких векторов отражения является артефактом реализации, зависящим от многословности сообщений об ошибках и других факторов. Но в IMAP каждая команда должна начинаться с так называемого тега, который должен быть отражен, чтобы клиент мог сопоставить ответ сервера на выданную команду. Хотя этот вектор отражения предписан стандартом протокола, допустимый набор символов может отличаться в разных реализациях.

Отраженные ответы, скорее всего, содержат некоторый "шум" до и после отраженной полезной нагрузки. В этом случае браузер должен поддерживать контент-сниффинг, чтобы разрешить отраженную XSS-атаку.

Атаки на загрузку и выгрузку данных по FTP. FTP использует два отдельных соединения для команд и данных. Таким образом, атакующий MitB запускает два запроса в браузере. Первый запрос

изменяет состояние FTP-сервера таким образом, что он открывает порт данных для клиента, чтобы загрузить или выгрузить файл. Хотя сервер возвращает клиенту номер порта данных, этот ответ хранится в контексте браузера целевого веб-сервера и недоступен для злоумышленника. Таким образом, атакующему приходится перебором подбирать нужный порт данных на сервере, как при атаке захвата портов. Затем злоумышленник запускает второй запрос в браузере и перенаправляет его на порт данных. При атаке на загрузку полный HTTP-запрос, включая все секретные файлы cookie в заголовке, хранится на FTP-сервере, где злоумышленник имеет доступ на чтение. При атаке на загрузку злоумышленник сначала готовит правильный HTTP-ответ с вредоносной полезной нагрузкой JavaScript и сохраняет его на FTP-сервере. При атаке на загрузку ответ возвращается клиенту.

Для атак на загрузку ответ FTP-сервера на соединение для передачи данных не содержит никаких протокольных помех, поэтому он работает в любом браузере независимо от наличия контент-сниффинга.

Атаки на выгрузку электронной почты. SMTP и IMAP могут использоваться для отправки или сохранения электронных писем на контролируемую злоумышленником учетную запись электронной почты и, таким образом, подходят для атак выгрузки. POP3 не поддерживает загрузку пользовательских данных.

Для SMTP злоумышленник MitB может вызвать в браузере запрос на вход в учетную запись злоумышленника на сервере и начать отправку электронного письма в эту учетную запись. Для IMAP запрос входит в учетную запись злоумышленника на IMAP-сервере и сохраняет черновик письма в контролируемой злоумышленником папке. В любом случае, первоначальный запрос подготавливает сервер к состоянию, в котором, если он получит второй запрос браузера, использующий то же соединение, содержимое всего запроса (включая cookie в заголовке) будет передано злоумышленнику.

Следует обратить внимание на то, что два запроса, инициированные атакующим MitB, используют не одно и то же, а разные TLS-соединения, поэтому FTP-атаки на загрузку работают независимо от повторного использования соединения в браузере.

Атаки на загрузку электронной почты. IMAP и POP3 могут использоваться для загрузки электронных писем с контролируемой злоумышленником учетной записи электронной почты и, таким образом, подходят для атак на загрузку. SMTP не поддерживает загрузку данных.

Для IMAP и POP3 злоумышленник MitB может вызвать в браузере запрос, содержащий команды для входа в аккаунт злоумышленника на сервере, выбрать почтовый ящик и получить письмо, содержащее

вредоносную полезную нагрузку, ранее сохраненную там злоумышленником. Ответ сервера IMAP или POP3 будет содержать содержимое всего письма, включая тело письма с вредоносной полезной нагрузкой.

Ответы серверов IMAP и POP3 включают всю транзакцию запроса, включая баннер сервера, любые сообщения об ошибках, ответы на команды login и другие команды, которые предшествуют содержимому загруженного письма. На практике атаки на скачивание электронной почты успешны только в том случае, если браузер жертвы поддерживает функцию контент-сниффинга.

3. Блокирование портов

В качестве обходного пути для противодействия ранним кросс-протокольным атакам с 2001 года браузеры блокируют доступ к определенным известным портам.

Как и ожидалось, большинство браузеров блокируют доступ к портам, используемым SMTP, IMAP, POP3 и FTP, за некоторыми исключениями. Например, порт 990 не заблокирован ни в одном из протестированных браузеров, поэтому кросс-протокольные атаки с использованием FTPS все еще возможны. Кроме того, Edge Legacy и Internet Explorer не блокируют порты 465 (SMTPS) и 995 (POP3S), что позволяет проводить кросс-протокольные атаки, использующие эти службы. С другой стороны, эти браузеры блокируют доступ к открытым (не-TLS) вариантам этих протоколов на портах 25 (SMTP) и 110 (POP3).

Поэтому из-за блокировки портов атаки не должны работать в сценарии чистого веб-атакера, если сервис не работает на нестандартном порту. На практике это не является нереальным, поскольку службы часто развертываются на нестандартных портах по различным административным причинам.

Еще одним препятствием, с которым приходится иметь дело в модели чистого веб-атакующего, являются межсайтовые ограничения, связанные с политикой одинакового происхождения (SOP), включая политику cookie . Доступ к DOM из одного места происхождения (идентифицируемого кортежем protocol: //host: port) в другое место происхождения не разрешен. Однако Edge Legacy и Internet Explorer игнорируют номера портов в SOP. Например, host:995 (POP3S) имеет доступ к host:443, что позволяет манипулировать DOM (т.е. читать или писать содержимое сайта, вставлять теги сценариев и т.д.). Более того, существуют технологии, такие как CORS, позволяющие пробивать дыры в SOP.

SOP является более мягким, когда речь идет о cookies. Как указано в RFC 6265, cookies не зависят от порта. Например, host:995 может

получить доступ к файлам cookie для host:443 во всех протестированных браузерах. Доступ к файлам cookie для субдоменов возможен только в том случае, если явно установлен флаг Domain.

Политика установки cookie еще менее строга, поскольку поддомену разрешается устанавливать cookie для верхнего домена. Например, pop3. host: 99 S может установить cookie для host:443 во всех браузерах, что может привести к атакам фиксации сеанса, когда злоумышленник блокирует пользователя в сеансе, уже контролируемом атакующим, еще до того, как пользователь вошел в систему.

Заключение

Таким образом, был оценён реальный ущерб от атаки веб-браузеров и широко распространенных серверов электронной почты и FTP. Установлены методы атаки веб-серверов с использованием уязвимого сервера приложений. Проанализирована эффективность расширений TLS, таких как Application Layer Protocol Negotiation (ALPN) и Server Name Indiciation (SNI), для смягчения этих и других межпротокольных атак.

Список литературы

1. Диченко, С. А. Концептуальная модель обеспечения целостности информации в современных системах хранения данных / С. А. Диченко // Информатика: проблемы, методология, технологии. Сборник материалов XIX международной научно-методической конференции. Под ред. Д. Н. Борисова. – Воронеж, 2019. – С. 697-701.

2. Горбачев, И. Е. Особенности обеспечения ИБ критической инфраструктуры с учетом специфики АСУ ТП / И. Е. Горбачев, Р. В. Лукьянов, А. М. Сухов // Защита информации. Инсайд. – 2016. – № 2 (68). – С. 30-37.

3. Dichenko, S. Two-dimensional control and assurance of data integrity in information systems based on residue number system codes and cryptographic hash functions / S. Dichenko, O. Finko // Integrating Research Agendas and Devising Joint Challenges International Multidisciplinary Symposium ICT Research in Russian Federation and Europe. – 2018. – P. 139-146.

4. Диченко, С. А. Контроль и восстановление целостности данных в защищенных информационно-аналитических системах / С. А. Диченко, О. А. Финько // Труды Военно-космической академии имени А.Ф.Можайского. – 2021. – № 676. – С. 36-49.

5. Диченко, С. А. Контроль и восстановление целостности многомерных массивов данных посредством криптокодовых

конструкций / С. А. Диченко, О. А. Финько // Программирование. – 2021. – № 6. – С. 3-15.

6. Сухов, А. М. Алгоритм применения методов и моделей противодействия компьютерным вторжениям / А. М. Сухов, С. В. Калиниченко, В. И. Якунин // Защита информации. Инсайд. – 2016. – № 6 (72). – С. 38-41.

7. Казарин, И. С. Обзор сетевых атак на информационные системы / И. С. Казарин, Е. М. Михайлова // В сборнике: Интеллектуальный потенциал XXI века: ступени познания. Сборник материалов XXXIX Молодежной международной научно-практической конференции. Под общей редакцией С.С. Чернова. – 2017. – С. 140-148.

8. Сухов, А. М. Методика моделирования процесса функционирования системы обнаружения вторжений в компьютерную сеть в задачах исследования эффективности / А. М. Сухов, И. Е. Горбачев, В. И. Якунин // Проблемы информационной безопасности. Компьютерные системы. – 2017. – № 2. – С. 23-30.

9. Диченко, С. А. Безопасные генераторы псевдослучайных линейных последовательностей на арифметических полиномах для защищенных систем связи / С. А. Диченко, О. А. Финько // Нелинейный мир. – 2013. – Т. 11. – № 9. – С. 632-645.

10. Сачков, И. К. Автоматизация противодействия бот-атакам / И. К. Сачков, А. Н. Назаров // T-Comm: Телекоммуникации и транспорт. – 2014. – Т. 8. – № 6. – С. 5-9.

11. Диченко, С. А. Снижение вводимой избыточности при обеспечении устойчивости информационно-аналитических систем в условиях компенсации последствий деструктивных воздействий злоумышленника / С. А. Диченко, О. А. Финько // Автоматизация процессов управления. – 2020. – № 4 (62). – С. 38-48.

12. Дементьев, В. Е. Понятийный аппарат протокольной защиты информационно-телекоммуникационной сети / В. Е. Дементьев, А. В. Дементьева, Д. А. Маняшин // В сборнике: Актуальные проблемы инфотелекоммуникаций в науке и образовании. Сборник научных статей. – 2016. – С. 70-74.