Научная статья
Original article
УДК 004.424

ВРЕДОНОСНЫЕ ЗАПРОСЫ

MALICIOUS INQUIRIES

Казарян Каторина Камоевна студент специалитета, Донской государственный технический университет, г. Ростов-на-Дону (344003 Россия г. Ростов-на-Дону, Гагарина 1), kazaryan_katorina@mail.ru

Белан Виктория Вадимовна магистрант, Донской государственный технический университет, г. Ростов-на-Дону, г. Ростов-на-Дону (344003 Россия г. Ростов-на-Дону, Гагарина 1), viktoriyabelan2018@gmail.com

Kazaryan Katorina Kamoevna specialty student, Don State Technical University, Rostov-on-Don (344003 Russia, Rostov-on-Don, Gagarina 1), kazaryan_katorina@mail.ru

Belan Victoria Vadimovna Master student, Don State Technical University, Rostov-on-Don, Rostov-on-Don (344003 Russia, Rostov-on-Don, Gagarina 1), viktoriyabelan2018@gmail.com

Аннотация. В данной статье описан анализ запросов на внутренний сервер. При анализе запросов можно инициализировать такие типы, которые могут принести вред различной тяжести пользователям и серверу в целом. Встречаются ситуации, когда внутренний сервер получает запросы различного рода. При работе он видит, что тело запроса очень короткое, всего 3 байта в длину. Сервер не обрабатывает запросы от MALICIOUS-REQUEST и

аналогичные. Вместо этого он рассматривает это как следующий запрос. Это заставляет сервер начинать обработку злонамеренного запроса как есть.

Abstract. This article describes the analysis of requests to the internal server. When parsing requests, you can initialize types that can bring harm of varying severity to users and the server as a whole. There are situations when the internal server receives requests of various kinds. When running, it sees that the request body is very short, only 3 bytes long. The server does not process requests from MALICIOUS-REQUEST and similar. Instead, it treats it as the next request. This causes the server to start processing the malicious request as is.

Ключевые слова: вредоносные запросы, обфускации заголовка, атака сервера

Keywords: malicious requests, header obfuscation, server attack

Введение

Уязвимости поведения кодирование-передача данных

Существуют следующие данные, например интерфейсный, и внутренний серверы правильно определяют приоритет заголовка Transfer-Encoding. Однако злоумышленник может скрыть заголовок, чтобы обмануть один из серверов.

Вот несколько способов обфускации заголовка. Большинство методов используют парные заголовки или заголовки Transfer-Encoding, один из которых не соответствует обычному соглашению Рисунок 1.

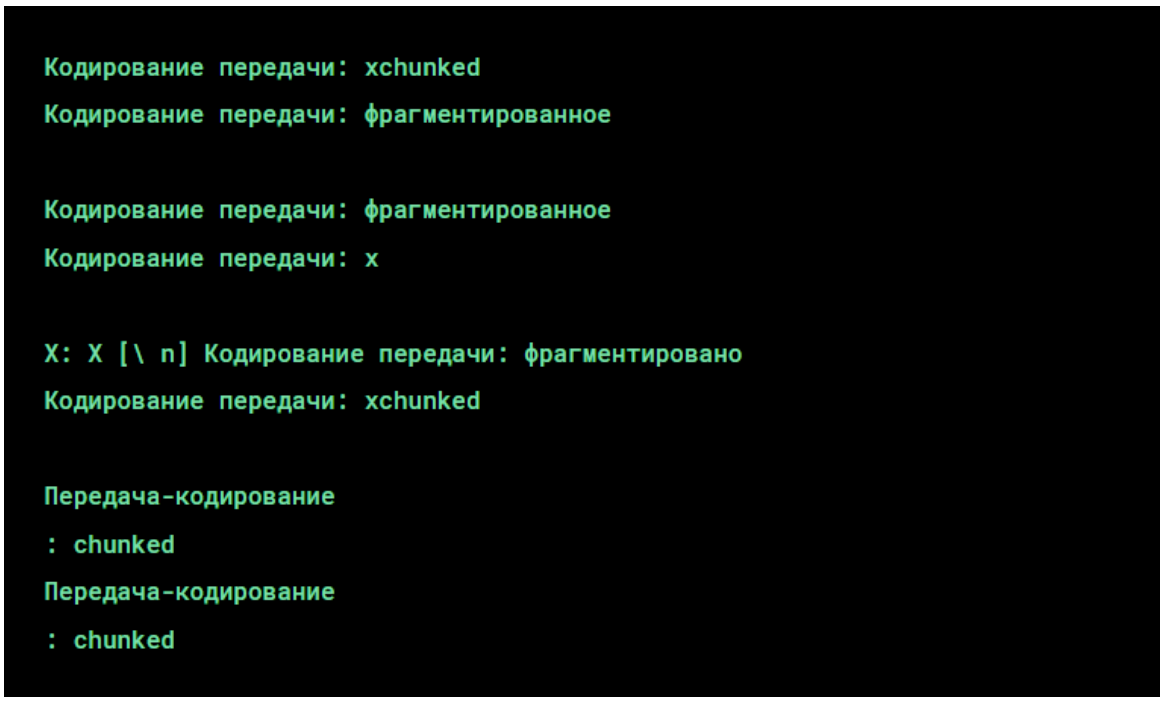


Рисунок 1 – Пример уязвимости поведения кодирование-передача данных

Поскольку этот код лишь незначительно отличается от спецификации HTTP, многие реализации сервера по-прежнему будут принимать их как допустимые. Для успешной атаки злоумышленникам необходимо найти вариант заголовка Transfer-Encoding, который будет обрабатываться одним из серверов, но игнорироваться другим.

Воздействие атаки будет зависеть от того, является ли внешний или внутренний сервер сервером, который обманом не обрабатывает заголовок Transfer-Encoding. В остальном атака будет похожа на CL.TE или TE.CL.

Расширенные атаки контрабанды HTTP-запросов

Обход фильтров безопасности

Этот тип атаки включает передачу вредоносного запроса непосредственно на внутренний сервер таким образом, чтобы он не обнаруживался фильтрами безопасности промежуточного программного обеспечения. Затем запрос выполняется на внутреннем сервере.

Замена штатного ответа

Этот тип атаки может использоваться, когда промежуточным программным обеспечением является кэш-сервер. Злоумышленник пытается выполнить отравление кеша, при котором недопустимые ответы сохраняются в записях кеша. После успеха первоначальной атаки запросы на будущее использование будут возвращать вредоносный запрос, который теперь хранится в кеше. Это может привести к отказу в обслуживании на сервере.

Взлом учетных данных

В этом типе атаки злоумышленник вводит часть запроса в поток запросов и ждет законного запроса конечного пользователя. Злоумышленник принимает запрос пользователя и добавляет его к частичному запросу, используя то же соединение. Прокси-сервер не знает, что это два отдельных запроса, рассматривая их как один.

Этого сложно добиться, но в случае успеха злоумышленнику удается «спрятать» поверх действующего сеанса пользователя , включая его куки и детали аутентификации HTTP. Они могут использовать эти достоверные данные о подключении, чтобы переправить свой злонамеренный запрос.

Это форма взлома учетных данных. При использовании в сочетании с контрабандой HTTP-запросов это может быть очень опасным, потому что злоумышленник может выполнять действия POST от имени пользователя, используя его учетные данные и уровень привилегий.

Как уменьшить уязвимость, связанную с контрабандой HTTP-запросов

Атаки с использованием HTTP-контрабанды являются изощренными, поскольку они используют неоднозначность интерпретаций и конфигураций протоколов сервера.

ИТ-специалисты могут рассмотреть несколько вариантов уменьшения подверженности этой уязвимости:

– Последовательная интерпретация заголовков HTTP на интерфейсных и внутренних серверах - очевидно, что это лучший вариант предотвращения. Однако это не всегда вариант, поскольку балансировщики нагрузки, как правило, представляют собой аппаратные устройства,

поддерживающие внутренние серверы, работающие на разных платформах. Если вы не можете запустить одно и то же программное обеспечение как на интерфейсной, так и на внутренней стороне, вы должны, по крайней мере, знать, как каждый сервер работает с заголовками HTTP, и убедиться, что они последовательно интерпретируют HTTP-запросы.

– Отключите уязвимые оптимизации - если вы не можете изменить конфигурацию серверной части, отключите любые оптимизации производительности, которые используют заголовок Transfer-Encoding или Content-Length. Это может снизить эффективность веб-среды, но очень эффективно предотвращает эту атаку.

– Избегайте использования балансировщиков нагрузки, сетей доставки контента (CDN) или обратных прокси-серверов - если эти компоненты не требуются в вашей настройке, вы обычно защищены от незаконной передачи HTTP-запросов.

– Использовать HTTP / 2 - обеспечение того, чтобы интерфейсный и внутренний серверы обменивались данными только с использованием протокола HTTP / 2, можно предотвратить большинство вариантов этой атаки.

– Отключите повторное использование соединения на внутреннем сервере - если это технически возможно в вашей среде, это может полностью предотвратить незаконную пересылку HTTP-запросов.

– Настройте интерфейсный сервер для нормализации неоднозначных запросов - это может предотвратить передачу злонамеренных запросов на внутренний сервер.

– Не раскрывайте зарегистрированный HTTP-трафик - журналы HTTP должны быть доступны только администраторам, чтобы избежать раскрытия непреднамеренных частей HTTP-запроса потенциальным злоумышленникам.

– Используйте брандмауэр веб-приложений (WAF) - многие технологии WAF определяют, блокируют или очищают HTTP-трафик, например, для запроса директив контрабанды. Организации, которые в

настоящее время используют WAF, должны подтвердить у поставщика, что уровень защиты активирован. Также важно проверить, требуются ли какие-либо изменения в конфигурации WAF для защиты от уязвимости, связанной с контрабандой HTTP-запросов.

В настоящее время очень многие пользователи сталкиваются с такой проблемой, как не инициализируемые запросы на компьютерные сервера. Не каждый пользователь имеет точное представление о том, что это такое и как с ним бороться.

На сегодняшний день пользователь может защитить компьютер от таких видов атак, соблюдая основные правила: пользоваться аналитическими программами, использовать наиболее новые версий ОС, редко работать за персональным компьютером под правами администратора, проверять надежность источников скачиваемой информации.

Литература

1. «Объём HTTP-трафика впервые превысил P2P» - net.compulenta.ru/322974/ — Компьюлента, 22 июня 2007. (Официальный документ от Ellacoya Networks - www.ellacoya.com/news/pdf/2007/NXTcommEllacoyaMediaAlert.pdf)
2. «HTTP-NG — объектно-ориентированный протокол передачи гипертекста» - www.osp.ru/cw/1998/20/29525/, «Открытые системы», 29 мая 1998 г.
   «(Proposed) HTTP-NG Working Group» - www.w3.org/Protocols/HTTP-NG/ — официальная страница W3C по разработке протокола HTTP-NG.
3. HTTP/1.1: Method Definitions - www.w3.org/Protocols/rfc2616/rfc2616-sec9.html (англ.). W3C.
4. См. первое предложение раздела «6.1.1 Status Code and Reason Phrase - tools.ietf.org/html/rfc2068#section-6.1.1» в RFC 2068. На стр. 40 - tools.ietf.org/html/rfc2068#page-40 есть также объявление в формате расширенной БНФ-формы (Augmented BNF) «extension-code = 3DIGIT» для кодов расширений.

5. HTTP errors - www.flickr.com/photos/apelad/sets/72157594388426362/detail/

References

1. "The volume of HTTP traffic exceeded P2P for the first time" - net.compulenta.ru/322974/ - Compulenta, June 22, 2007. (White paper from Ellacoya Networks - www.ellacoya.com/news/pdf/2007/NXTcommEllacoyaMediaAlert.pdf)
2. "HTTP-NG - Object-Oriented Hypertext Transfer Protocol" - www.osp.ru/cw/1998/20/29525/, "Open Systems", May 29, 1998
   The (Proposed) HTTP-NG Working Group - www.w3.org/Protocols/HTTP-NG/ is the official W3C HTTP-NG protocol development page.
3. HTTP / 1.1: Method Definitions - www.w3.org/Protocols/rfc2616/rfc2616-sec9.html (English). W3C.
4. See the first sentence of "6.1.1 Status Code and Reason Phrase - tools.ietf.org/html/rfc2068#section-6.1.1" in RFC 2068. On page 40 - tools.ietf.org/html/ rfc2068 # page-40 there is also an Augmented BNF "extension-code = 3DIGIT" declaration for extension codes.
5. HTTP errors - www.flickr.com/photos/apelad/sets/72157594388426362/detail/

© Казарян К.К., Белан В.В., 20221 Научно-образовательный журнал для студентов и преподавателей «StudNet» №1/2022.

Для цитирования: Казарян К.К., Белан В.В. ВРЕДОНОСНЫЕ ЗАПРОСЫ // Научно-образовательный журнал для студентов и преподавателей «StudNet» №1/2022.