АТАКА FREAK НА TLS/SSL

Андросова Татьяна Евгеньевна

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Курочкин Владислав Михайлович

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Болдырев Артем Сергеевич

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Чернов Роман Вячеславович

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Введение.

В январе 2015 года проект OpenSSL опубликовал предупреждение о том, что его клиенты успешно приняли слабый export-ключ RSA с сервера во время полномасштабного установления соединения (handshake). Крайне важно, что уязвимость существовала даже в том случае, если клиент не предлагал использовать какие-либо экспортные шифронаборы с RSA. Эта уязвимость была классифицирована как CVE-2015-0204 и изначально не привлекала большого внимания.

Некоторые исследователи поняли потенциал новой уязвимости и начали работать над реальной атакой. В начале марта они объявили, что успешно использовали CVE-2015-0204 для MITM-атаки на веб-сайт *www.nsa.gov*. Они также дали атаке имя: FREAK, что является аббревиатурой для *Factoring RSA Export Keys* (факторизация export-ключей RSA).

Цель состояла в том, чтобы показать, что большое количество серверов в Интернете уязвимы для практических атак MITM. В результате работы исследователей первоначальное открытие было переклассифицировано из низкой в высокую степень угрозы.

Export-криптография.

Перед тем, как начать обсуждать FREAK, нам нужно вернуться в 1990-е года, и понять, что такое *экспортные шифронаборы*. До сентября 1998 года в США действовал закон об ограничении экспорта стойких шифров за пределы страны, ограничивая длину ключа для симметричного шифрования до 40 бит и до 512 бит – для асимметричного. Экспортные наборы были разработаны специально для того, чтобы придерживаться этих ограничений.

Экспортные комплекты шифров начали устаревать после того, как США смягчили экспортный контроль криптографии в январе 2000 года, но код остался во многих (большинстве?) библиотек SSL / TLS. Как это бывает, некоторые вещи из этого старого кода могут быть вызваны манипулированием сообщениями протокола.

Атака.

Во время обычного обмена ключами RSA клиент генерирует случайный premaster secret**,** зашифровывает его открытым ключом RSA сервера и отправляет на сервер. Если ключ RSA сильный, обмен ключами также силен. Когда согласовывается экспортный шифронабор, сервер генерирует слабый ключ RSA, подписывает его своим сильным ключом и отправляет слабый ключ клиенту в сообщении ServerKeyExchange. Затем клиент использует этот слабый ключ для шифрования premaster secret, чтобы соответствовать export-правилам. Несмотря на то, что ключ слабый, злоумышленник в сети не может использовать его для активной атаки, потому что подпись по-прежнему сильна (при условии, что ключ сервера будет сильным, конечно).

На сегодняшний день экспортные шифронаборы стали безнадежно слабы. Хотя злоумышленник в сети не может вмешиваться в процесс установления соединения, он может записать все передаваемые сообщения, а затем методом перебора вычислить слабый ключ, восстановить premaster secret и расшифровать все остальное. Атакующий с большими вычислительными ресурсами может сделать это за минуты или даже быстрее. Практически каждый может сделать это в течение нескольких часов.

К счастью, современные клиенты больше не поддерживают экспортные наборы, но FREAK является мощной атакой, потому что он не нуждается в них. Во время обычного обмена ключами RSA сообщение ServerKeyExchange не разрешено. Но все же, уязвимые библиотеки все еще обрабатывают его, а затем используют поставляемый слабый ключ RSA для обмена ключами.

Чтобы использовать FREAK, злоумышленник должен каким-то образом навязать сообщение ServerKeyExchange жертве. Это уменьшит надежность соединения только до 512 бит. Однако есть два препятствия: (1) введенная подпись должна быть подписана сильным ключом RSA, используемым на целевом сервере, и (2) вмешавшись в процесс установления соединения TLS, злоумышленник должен найти способ также подделать правильное сообщение Finished, чтобы узаконить изменения, даже если это то, что должен делать только реальный сервер. Из-за первого препятствия атака работает только на серверах, поддерживающих экспортные шифронаборы.

Злоумышленник напрямую подключается к серверу, предлагая только сам экспортный шифронабор. Это приводит к успешному согласованию этого набора, после чего сервер отправляет сообщение ServerKeyExchange. Трюк здесь состоит в том, чтобы повторно использовать это сообщение против жертвы. Несмотря на то, что TLS защищает от атак подписи, защита зависит от случайных чисел клиента и сервера (ClientRandom и ServerRandom), отправленных в ClientHello и ServerHello соответственно. Однако активный атакующий в сети может дождаться, когда клиент будет подключаться первым, а затем скопирует исходные случайные числа в отдельное соединение с целевым сервером. Конечным результатом является сообщение ServerKeyExchange, которое проходит проверку у жертвы.

Большей проблемой является создание правильного сообщения Finished. Это сообщение зашифровано и содержит хэш всех сообщений, передаваемых в процессе установления соединения. Обе стороны во время обмена подтверждают, что содержимое этого сообщения соответствует их собственному расчету. Злоумышленник не может просто изменить эти сообщения из-за проверки шифрования и целостности. Однако к настоящему времени злоумышленнику удалось уменьшить стойкость ключей до 512 бит. Если у него есть доступ к мощным вычислительным мощностям, он может вычислить этот слабый ключ методом прямого перебора в реальном времени, расшифровать premaster secret, отправленный клиентом, и получить полный контроль над соединением. Это также означает возможность отправки правильных сообщений Finished.

Теперь 512-битные ключи слабы, но не настолько. Разумеется, нам следует ожидать, что некоторые организации способны вычислять такие ключи в режиме реального времени, но это не то, о чем все должны беспокоиться. Однако становится все хуже. По соображениям производительности вместо генерирования слабых ключей для каждого нового подключения некоторые серверы генерируют только один слабый ключ и повторно используют его в

течение определенного периода времени. Иногда он используется долгое время, достаточного для того, чтобы даже атакующий с маленькими вычислительными ресурсами смог вычислить его. Вот что сделали исследователи. Они идентифицировали сервер, который повторно использовал ключи, потратили около $100 на ресурсы облачных вычислений и вычислили ключ примерно за семь часов. С помощью ключа исследователи могли тривиально выполнять атаки MITM против всех уязвимых клиентов до тех пор, пока ключ оставался неизменным.

Способы защиты.

Первоначально считалось, что только OpenSSL уязвим для атаки FREAK. Хотя немногие браузеры используют OpenSSL для клиентских операций, один большой пользователь – платформа Android, что означает, что миллиарды телефонов потенциально уязвимы. Эта проблема затрагивает не только браузеры, но, вероятно, также все приложения, запущенные на уязвимых платформах.

Совет для операторов серверов заключается в удалении наборов экспортных шифров, поскольку они необходимы для проведения атаки, но многие операторы небрежно консервативны при удалении устаревших функций протокола.

Список литературы:

1. Ristic I. Bulletproof SSL and TLS. – London: Feisty Duck Limited, 2015. – 516 c.

2. The FREAK Attack – Censys – [Электронный ресурс] – Режим доступа. – URL: https://censys.io/blog/freak (дата обращения: 13.06.2017).