ОТЧЕТ
ПО РЕЗУЛЬТАТАМ НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЫ

ИСПОЛЬЗОВАНИЕ СИСТЕМЫ DNS СОВРЕМЕННЫМИ СИСТЕМАМИ ОБМЕНА МГНОВЕННЫМИ СООБЩЕНИЯМИ

Декабрь 2017 года

ВВЕДЕНИЕ

Предметом исследования являлись сценарии использования системы и службы доменных имён (DNS) приложениями мессенджеров, предназначенными для смартфонов. Под мессенджерами подразумеваются «системы мгновенного обмена сообщениями».

В список мессенджеров, выбранных для исследования, вошли наиболее распространённые (и узнаваемые) в России решения:

- Telegram;
- Facebook Messenger;
- WhatsApp;
- Viber.

Для обеспечения передачи сообщений, приложения мессенджеров устанавливают соединения с внешними серверами, принадлежащими провайдеру сервиса. Определение IP-адресов этих серверов может проводиться с использованием DNS. Такой метод считается предпочтительным, так как позволяет приложениям эффективно и в прозрачном режиме адаптироваться к изменению настроек адресов. Необходимо отметить, что, кроме типового использования DNS, возможны и другие решения, также обладающие определённой гибкостью. В том числе, возможны варианты с распространением списков IP-адресов серверов непосредственно в составе приложения. Из исследованных приложений - только Telegram не использует DNS явным образом: по крайней мере, сигнатуры DNS в трафике приложения отсутствуют. Очевидно, что данное приложение содержит встроенные списки адресов.

ТЕСТОВАЯ СРЕДА И ИСПОЛЬЗОВАННЫЕ МЕТОДЫ

В рамках настоящего исследования анализировались только мобильные приложения на платформе Android версии 7.0 (современная линейка). Веб-клиенты, а также клиенты для настольных компьютеров - не рассматривались. В качестве тестовой платформы для приложений использовался смартфон Samsung Galaxy J3 с заводскими настройками.

Метод исследования состоит в перехвате сетевого трафика, генерируемого и принимаемого смартфоном, на котором исполняются тестируемые приложения. Для перехвата трафика было создано тестовое окружение, состоявшее из точки доступа WiFi, работающей на базе одноплатного компьютера Raspberry Pi, подключенного к Интернету,

и настольного компьютера, получавшего с Raspberry Pi копию трафика беспроводного порта. В состав ПО точки доступа входил специально настроенный DNS-сервер (BIND 9), с логированием запросов. Данный DNS-сервер служил рекурсивным резолвером, его адрес передавался в ОС смартфона штатным образом, через DHCP. Трафик беспроводного интерфейса точки доступа анализировался при помощи программного пакета Wireshark (линейки версий 1.12). Тестовое устройство (смартфон) подключалось к WiFi-сети лабораторной точки доступа, далее на смартфоне запускались приложения мессенджеров и осуществлялись тестовые действия: приём и отправка сообщений различных форматов. Таким образом, никакого вмешательства в штатную работу приложений не производилось, а сам сценарий полностью соответствует обычной схеме использования WiFi (нередко владельцы смартфонов подключаются к публичным точкам доступа, где используют локальный рекурсивный резолвер).

Использовался и анализировался только трафик IPv4. Приложения использовались в типовой настройке «из коробки», в некоторых случаях проводились дополнительные настройки - такие случаи отмечены в описании отдельно.

Тестовые сценарии (подмена DNS) реализовывались путём размещения непосредственно на резолвере локальных DNS-зон, соответствующих зонам технических имён, используемых приложениями. Перечень таких имён определялся при помощи анализа трафика DNS-запросов. Для определения способа обработки ссылок, а также детектирования возможных подключений к «подменным» IP-адресам - использовался внешний виртуальный веб-сервер.

РЕЗУЛЬТАТЫ

Telegram (версия v4.5.1 (1144) arm-v7a)

Данное приложение не опрашивает DNS штатным образом. Возможно, что DNS используется опосредовано, через центральные серверы. Telegram обращается к этим серверам непосредственно по IP-адресам, перечень которых входит в состав приложения. Используется несколько IP-адресов, по всей вероятности, соответствующих разным дата-центрам: такой вывод можно сделать из опубликованного исходного кода приложения.

При передаче ссылок (URL) в сообщениях, Telegram обращается к DNS, определяя адрес (A-запись) для имени хоста, входящего в состав ссылки, а также, при обнаружении A-записи, выполняет GET-запрос по передаваемому адресу. GET-запрос выполняется серверами Telegram, клиентское устройство при этом не задействовано.

Пример запроса:
149.154.167.171 - "GET / HTTP/1.1" 200 77 "-" "TelegramBot (like TwitterBot)"

Однако, опрос DNS с целью определения адреса по имени хоста, указанного в URL, проводится через локальный резолвер с устройства (функция включена по умолчанию для обычных чатов; для секретных - выводится отдельное окно с предупреждением).

Статус подмены DNS в Telegram

Приложение не использует DNS для обнаружения каких-то значимых сервисов, поэтому подмена ответов не имеет смысла. Тем не менее, включённая по умолчанию для обычных чатов функция «предпросмотра ссылок» может приводить к утечке информации о локальном IP на произвольные внешние серверы имён (для этого пользователь должен отправить кому-то ссылку, содержащую заданную доменную зону в имени).

Facebook messenger (версия 146.0.0.33.136)

Приложение использует DNS для определения актуальных адресов серверов. Обнаруженные имена находятся в зонах второго уровня facebook.com (основные сервисы), fbcdn.net (используется, в частности, при отправке изображений), fbsbx.com (при работе в режиме видеозвонка).

Выявленные в процессе анализа трафика имена (во всех случаях - запрашиваются A-записи; содержательным ответом обычно является CNAME, см. таблицу 1):

graph.facebook.com.
api.facebook.com.
connect.facebook.com.
*.fna.fbcdn.net. (блок адресов CDN, пример представителя: scontent.fhel5-1.fna.fbcdn.net.)
stun.fbsbx.com.
rupload.facebook.com.

Запрос	Штатный ответ (фрагмент)
graph.facebook.com., A?	CNAME api.facebook.com
api.facebook.com., A?	CNAME star.c10r.facebook.com
star.c10r.facebook.com., A?	A 31.13.72.8
connect.facebook.com., A?	CNAME www.facebook.com
scontent.fhel5-1.fna.fbcdn.net., A?	A 81.27.242.17

Таблица 1. Примеры запросов и ответов DNS для Facebook Messenger

Интересно, что в случае с зоной fna.fbcdn.net. ответы получены с IP-адресом (81.27.242.17), принадлежащим российской автономной системе. Маршрутная информация данной AS: https://www.ididb.ru/autnum/#AS20764.

Подмена всех ответов по указанным зонам на IP-адрес 0.0.0.0 приводит к тому, что работа приложения блокируется (подмена отдельных ответов - к сбоям доставки сообщений).

При этом, после обнаружения в ответе IP-адреса, по которому соединение заведомо невозможно (0.0.0.0), приложение некоторое время использует «старые» адреса, вероятно, сохранённые в кеше. При перезагрузке устройства - «старые» адреса удаляются немедленно.

При подмене ответа IP-адресом подставного тестового сервера, приложение пытается установить TLS-соединение (443/tcp) по переданному в DNS-ответе IP-адресу. Это означает, что никакой защиты от подмены не используется (аналогичная ситуация - с другими приложениями, см. ниже).

В случае, если сервер-имитатор устанавливает TLS-соединение, но передаёт самоподписанный сертификат (заведомо не валидный), приложение завершает TLS Handshake с фатальной ошибкой («Неизвестный УЦ»). Такое поведение является корректным в контексте TLS, так как приложению не удалось аутентифицировать узел. Никаких дополнительных данных, кроме требуемых для начального установления соединения (Handshake), не передаётся.

При передаче ссылок внутри сообщений, Facebook Messenger проводит резолвинг имён и выполняет GET-запрос (при обнаружении A-записи). И опрос DNS, и проверка HTTP - производятся серверами Facebook Inc., дополнительных пользовательских данных, кроме (потенциально) содержащихся в составе URL, внешним серверам не передаётся.

Статус подмены DNS в Facebook Messenger

Подмена DNS полностью доступна, дополнительной проверки подлинности адресов (DNSSEC или другие методы) не производится. Приложение соединяется с сервером-имитатором, при этом, потенциально, происходит утечка следующей информации: факт использования Facebook Messenger; адреса узлов CDN, к которым производилось обращение (в том числе, в составе TLS Handshake); TLS-тикет и идентификатор TLS-сессии, унаследованные от успешного соединения с серверами Facebook. (Необходимо отметить, что данная информация штатно передаётся в открытом виде и может быть прочитана промежуточными узлами; однако, в случае с подменой DNS, TLS-данные могут быть раскрыты удалённому серверу, который не является промежуточным при доставке трафика; кроме того, данный сервер может, в теории, проксировать трафик, записывая его, - см. более подробный анализ в разделе «Комментарий».)

WhatsApp (версия: 2.17.395)

Приложение активно использует DNS, как для определения адресов серверов, так и во вспомогательных целях. Основная часть обнаруженных в трафике имён находится в зоне второго уровня whatsapp.net. Одно из служебных имён - в домене facebook.com (WhatsApp принадлежит Facebook Inc.).

Выявленные в процессе анализа трафика имена (во всех случаях - запрашиваются A-записи; как и в случае Facebook, содержательным ответом часто является CNAME, см. таблицу 2.):

e_NN_.whatsapp.net. (здесь e_NN_ - обозначает набор имён, выбираемых приложением, пример представителя: e10.whatsapp.net)
mmg-fna.whatsapp.net.
mmx-fb.cdn.whatsapp.net.
mmg-fna.whatsapp.net.
mqtt-mini.facebook.com.

Запрос	**Штатный ответ (фрагмент)**
e10.whatsapp.net., A?	A 169.55.67.240
mmg-fna.whatsapp.net., A?	CNAME mmx-fb.cdn.whatsapp.net
mmx-fb.cdn.whatsapp.net., A?	A 81.27.242.34

Таблица 2. Примеры запросов/ответов DNS для WhatsApp

Здесь также присутствует IP-адрес (81.27.242.34), принадлежащий российской AS.

WhatsApp выполняет предварительный опрос DNS при наборе пользователем URL в поле редактирования сообщения. Опрос производится через локальный резолвер. То есть, происходит утечка большого числа имён, составляющих ссылку. Например, при наборе example.com.gov.ru, приложение, следом за пользовательским вводом, будет пытаться резолвить имена example.local, example.com. example.com.gov, example.com.gov.ru. Для использования DNS не требуется отправлять сообщение со ссылкой.

В случае, если для какого-либо имени удалось определить значение A-записи, приложение делает попытку (HTTP) GET-запроса по соответствующему имени хоста. Данный запрос также выполняется с локального устройства, что приводит к раскрытию информации в сторону удалённого узла, на котором находится веб-сервер.

Пример запроса:
"GET / HTTP/1.1" 200 77 "-" "WhatsApp/2.17.395 A"

При подмене ответов DNS для всех имён *.whatsapp.net на 0.0.0.0 (A), приложение теряет возможность отправки и приёма сообщений. При этом сообщения, которые пользователь пытался отправить, отмечаются как отложенные, каких-то дополнительных сообщений о потере связи с сервисом не отмечено.

При подмене ответов на адрес сервера, с открытым доступом 80/tcp - приложение, после перезапуска устройства, отправляет по подменному адресу POST-запрос, содержащий некоторый блок данных (вероятно, запрос на установление соединение). То есть, при подмене адреса можно получить открытые HTTP-запросы в направлении

перехватывающего сервера (данные, передаваемые в запросе, не анализировались - они могут быть защищены). Пример запроса (из лога веб-сервера на подменном узле):

"POST /chat HTTP/1.1" 400 306 "-" "Mozilla/5.0 (compatible; WAChat/1.2; +http://www.whatsapp.com/contact)"

При замене ответов на адрес сервера, поддерживающего TLS (443/tcp), приложение так же прозрачно пытается соединиться с указанным IP-адресом, если ранее успешно использовалось для обмена сообщениями. Тестовый TLS-сервер возвращал самоподписанный сертификат (заведомо невалидный). Приложение WhatsApp корректно прерывало TLS-сессию («Неизвестный УЦ»), так как узел не проходил аутентификацию. При этом, в направлении удалённого узла, аналогично ситуации с Facebook Messenger, происходила утечка некоторых параметров прошлых TLS-соединений.

Статус подмены DNS в WhatsApp

Приложение полностью уязвимо для атак с подменой DNS: каких-либо проверок подлинности адресной информации (DNSSEC, другие методы) приложение не проводит. Эксперимент выявил, что приложение пытается соединяться с адресом подменного сервера, в том числе, по собственному REST-протоколу в открытом виде (POST-запросы). Приложение раскрывает в направлении внешних серверов IP-адрес пользовательской точки выхода в Интернет. В том числе, такое раскрытие происходит при наборе произвольных ссылок в поле редактирования сообщений, и при соединении с подменным IP-адресом, имитирующим сервер WhatsApp.

Viber (версия: 7.9.4.11)

Приложение активно использует DNS, в том числе, для определения адресов узлов сопутствующих сервисов, не относящихся к провайдеру именно сервиса обмена сообщениями. Данные внешние сервисы используются для сбора пользовательской статистики; такой сбор - активирован по умолчанию (при помощи ручных настроек его можно отключить, но в приложении в явном виде нигде не сообщается, что используются внешние сервисы - упоминание присутствует в Пользовательском соглашении).

Упомянутые только что внешние сервисы обнаружены под следующими именами:
api.mixpanel.com
app-measurment.com
e.crashlytics.com
cdn.chatleap.com

Все они полностью подвержены подмене DNS. Детально данные направления не рассматривались. Однако результат эксперимента показывает, что при подмене адресов данных сервисов, приложение пытается устанавливать TLS-соединения с подменными узлами.

Для именования системных сервисов Viber использует зону второго уровня viber.com. Все наблюдавшиеся имена - являются синонимами для фронтенд-узлов Amazon и Akamai. В трафике обнаружены следующие имена:

content.cdn.viber.com
secure.viber.com
aloha.viber.com
pg.cdn.viber.com
market.viber.com
media.cdn.viber.com
dl-media.viber.com
ads.viber.com

Примеры запросов даны в таблице 3.

Запрос	**Штатный ответ (фрагмент)**
content.cdn.viber.com., A?	CNAME d2jlvwphuziop7.cloudfront.net
secure.viber.com., A?	CNAME refugee-use-1985070560.us-east-1.elb.amazonaws.com
aloha.viber.com., A?	CNAME lb1-lbsn-1iav06yi7gaq2-976996164.us-east-1.elb.amazonaws.com
pg.cdn.viber.com., A?	CNAME www.viber.com.edgekey.net
market.viber.com., A?	CNAME www.viber.com.edgekey.net
media.cdn.viber.com., A?	CNAME do2gy2kwak9k2.cloudfront.net
dl-media.viber.com., A?	CNAME d1fje9gm3d05t8.cloudfront.net

Таблица 3. Примеры запросов/ответов DNS для приложения Viber

Приложение Viber также выполняет предпросмотр URL, вводимых пользователем (при отправке). При этом используется локальный резолвер DNS, а HTTP-запросы выполняются с локального устройства. Что, как и в случае с другими мессенджерами, приводит к утечке информации об IP-адресах и локальных настройках на пользовательском устройстве. Примеры запросов (Viber, в отличие от других исследованных приложений, выполняет два запроса: HEAD и GET):

"GET / HTTP/1.1" 200 77 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_4) AppleWebKit/537.36 (KHTML, like Gecko) QtWebEngine/5.6.0 Chrome/45.0.2454.101 Safari/537.36"
"HEAD / HTTP/1.1" 200 - "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_4) AppleWebKit/537.36 (KHTML, like Gecko) QtWebEngine/5.6.0 Chrome/45.0.2454.101 Safari/537.36"

Приложение подвержено атакам с подменой DNS. В случае замены всех ответов для *.viber.com. (A) на значение 0.0.0.0 - приложение теряет возможность отправки/получения сообщений и выводит пользовательское предупреждение «Разъединено».

При подмене на адрес сервера, отвечающего на TLS-соединения по 443/tcp, но с самоподписанным сертификатом, - приложение проводит TLS Handshake с именем сервера media.cdn.viber.com (или aloha.viber.com), после чего закрывает соединение с фатальной ошибкой, так как удалённый узел не проходит аутентификацию. Пользовательские данные не передаются. Попыток установить соединение на других номерах портов или отправить в открытом виде HTTP-запросы - выявить не удалось.

Статус подмены DNS в Viber

Приложение использует DNS не только для обнаружения адресов узлов, относящихся к сервису, но и для работы с дополнительными (внешними) сервисами статистики, а также для предпросмотра вводимых пользователем URL. Обращения проводятся с локального устройства. Все эти направления уязвимы для подмены ответов DNS, так как какая-либо проверка подлинности адресной информации не проводится (аналогично другим исследованным приложениями). Подмена DNS позволяет удалённым узлам фиксировать факт использования приложения Viber. Приложение раскрывает в направлении внешних серверов данные об IP-адресе точки выхода в Интернет пользователя.

КОММЕНТАРИИ

DNS является фундаментальным элементом современного Интернета. При этом, атаки, направленные на DNS, позволяют вынуждать пользовательские устройства к установлению тех или иных соединений с удалёнными серверами-имитаторами. Несмотря на то, что сам сервис DNS достаточно редко выступает в роли непосредственного транспорта для осуществления практических атак, отсутствие защиты от подмены адресов позволяет злоумышленникам действовать при помощи других, классических, инструментов, например, через стек TLS. Большинство эффективных атак на TLS требуют, чтобы клиентское устройство обратилось к атакующему серверу, подмена ответов DNS является мощным инструментом, позволяющим этого добиться. (В сравнении с активной подменой пакетов трафика, когда атакующий узел выдаётся за легитимный на уровне IP, подмена DNS является и более гибкой, и более универсальной, и несравнимо более простой для реализации схемой, позволяющей перенаправлять даже устройства, находящиеся в других сетевых сегментах.) Также, типовым направлением атак являются сами протоколы, используемые мессенджерами. Соответственно, перенаправление клиентов на подставные узлы - позволяет реализовать и такие атаки.

DNS является транспортом для утечек информации об активности пользователя. Три приложения из исследованных (Telegram, WhatsApp и Viber) - используют локальный DNS (на стороне пользователя) для реализации предпросмотра URL. Это означает, что злоумышленник, который методами социальной инженерии заставил пользователя

ввести в мессенджер тот или иной специально подготовленный URL, сможет получить, как минимум, сведения об IP-адресе точки выхода в Интернет (часто, этого достаточно для определения местоположения пользователя).

В случае, когда приложения устанавливают TLS-соединение с подменным узлом, куда были перенаправлены с помощью DNS, на данном узле может быть организован прокси-канал в сторону легитимных сервисов (практическая реализация не проводилась, так как находится за пределами настоящего исследования). Несмотря на то, что трафик, проходящий через прокси, будет защищён протоколом TLS, он может быть записан и расшифрован позже, когда атакующей стороне тем или иным способом станут известны секретные ключи.

ВЫВОДЫ

Исследованы сценарии использования DNS приложениями распространённых мессенджеров. Выявлено, что ни одно из приложений не проводит какой-либо проверки подлинности адресной информации, получаемой из DNS. При этом только Telegram, который не использует DNS для определения адресов технических узлов сервиса, оказался не уязвим к атакам подмены. Три других приложения (Facebook Messenger, WhatsApp, Viber) - подвержены атакам подмены адресной информации в DNS.

Стандартной рекомендацией для исправления ситуации является применение группы технологий DNSSEC, с внедрением валидации DNS-ответов на стороне клиента. Однако ни одна из технических зон, используемых исследованными приложениями, не поддерживает DNSSEC. Возможны и другие, существенно менее универсальные, методы проверки «подлинности» IP-адресов, полученных из DNS. В том числе, по принадлежности их к заданным блокам. Также, при внезапной смене IP-адреса, приложение может использовать дополнительные меры обеспечения безопасности: например, проверить, что действительно утрачена связь с только что использовавшимися адресами; запросить, в защищённом режиме (TLS), сведения об изменении IP-адресов.