Золотых М.О., версия 2023.04

СПЕЦИАЛЬНЫЕ ТЕХНИКИ АНАЛИЗА ТРАФИКА В WIRESHARK

Описание работы

Wireshark, благодаря множеству встроенных фильтров и функций, является очень эффективным инструментом при расследовании происшествий в сети. Данная работа направлена на изучение некоторых функций сетевого анализатора, которые могут быть полезны специалистам по информационной безопасности.

Во время выполнения работы закройте все лишние вкладки в используемых браузерах и старайтесь не запускать лишних приложений, генерирующих сетевой трафик. Это необходимо для того, чтобы упростить поиск необходимых пакетов. В процессе работы не забывайте сохранять захваченный трафик: он может повторно понадобиться при оформлении отчёта.

Данная работа, с небольшими изменениями, может быть выполнена в ОС семейства Linux.

Системные требования:

- компьютер с ОС Windows 10 или выше;
- подключение к Интернет (Ethernet или Wi-Fi с присвоением адреса по DHCP);
- программа Wireshark версии не ниже 2.6.0 (с установленным пакетом WinPCap), рекомендуется использование англоязычного интерфейса (т.к. он меньше подвержен изменениям);
- веб-браузер Mozilla Firefox версии не ниже 70;
- актуальные базы GeoLite от MaxMind для ASN, стран и городов;
- VMWare Workstation версии не ниже 15 с образом Ubuntu Server версии не ниже 18.04 LTS (необходимы пакеты tcpdump, links, traceroute, openssh и сетевое подключение в режиме bridge);
- WinSCP версии не ниже 5.15.

1. Просмотр TLS-трафика

Большинство современных веб-сайтов используют зашифрованное соединение для передачи данных. В большинстве случаев шифрование обеспечивается протоколом TLS (который заменил менее стойкий SSL). TLS работает между транспортным и прикладным протоколом (в случае веб-трафика – между TCP и HTTP). На наличие TLS указывает схема «*https*» в URL. В этом случае увидеть данные HTTP протокола напрямую в анализаторе трафика нельзя.

Для просмотра данных, инкапсулированных в TLS анализатору необходимо предоставить сеансовые ключи шифрования. В свою очередь шифрование осуществляет браузер, поэтому ключи шифрования находятся в нём.

1.1. Если браузер FireFox уже был запущен, то закройте все его окна. Затем запустите консоль операционной системы, в которой выполните команды: «**set SSLKEYLOGFILE=%HOMEDRIVE%%HOMEPATH%\keys.log**» (устанавливает переменную окружения для текущего сеанса командной строки), а затем «**“%ProgramFiles%\Mozilla Firefox\firefox.exe”**» (запускает браузер). Обратите внимание, что обе команды должны быть запущены из одной командной строки, так как команда **set** влияет только на текущий сеанс. Откройте в браузере любой сайт со схемой **https** и проверьте, что файл **keys.log** появился в домашней директории пользователя. Откройте этот файл и запишите в отчёт первую строку, содержащуюся в нём.

1.2. Запустите захват трафика и откройте в FireFox страницу **https://max.zolotyh.su/lab03** (обязательно явно укажите браузеру схему получения документа https). В открывшейся форме введите свой псевдоним в качестве имени пользователя и любую случайную комбинацию символов в качестве пароля, затем нажмите кнопку входа. Остановите захват трафика. Отфильтруйте трафик отправки данных на сайт по его IP-адресу, а затем отобразите содержимое TCP-потока. Зафиксируйте результат на снимке экрана. Можете ли вы понять, какая информация передавалась в потоке?

1.3. Откройте окно настроек Wireshark (меню «*Edit → Preferences…*»), затем в левой части окна выберите пункт «*Protocols*» и подпункт «*TLS*». На открывшейся вкладке в поле «*(Pre)-Master-Secret log filename*» укажите полный путь к файлу ключей, созданному на шаге 1.1, и закройте окно нажатием клавиши «*OK*». Уберите фильтр по TCP-потоку и верните фильтр по IP-адресу. Обратите внимание: в списке захваченных пакетов появились HTTP и другие заголовки, ранее скрытые, в декодере пакетов появились заголовки, вложенные в заголовок TLS, а в нижней части побайтового отображения пакета появилась закладка «*Decrypted TLS*», позволяющая просматривать расшифрованное содержимое без использования декодера полей. Зафиксируйте список захваченных пакетов на снимке экрана.

1.4. В списке захваченных пакетов найдите тот, у которого текст в столбце «*Info*» начинается с «*POST /lab03 …*». Вызовите контекстное меню для этого пакета и выберите пункт «*Follow → TLS Stream*». Зафиксируйте результат на снимке экрана. Найдите и отметьте на этом снимке имя пользователя и пароль, отправленные вами на сервер и перехваченные сетевым анализатором.

2. Работа с HTTP2 потоком

2.1. Запустите захват трафика и откройте в FireFox страницу **https://ya.ru**. В открывшейся поисковой форме введите любую комбинацию из латинских символов (во избежание проблем с выбором кодировки при дальнейшем анализе заголовков не используйте кириллицу) и нажмите кнопку «*Найти*». Остановите захват трафика. Аналогично предыдущему параграфу восстановите TLS-поток отправки поискового запроса. Обратите внимание, что поток не читаем для человека. Это происходит потому, что он закодирован в бинарном коде по протоколу HTTP2 (обратите внимание, что кодирование и шифрование – разные процессы). На это указывает «магическая» последовательность «*PRI * HTTP/2.0…*» в начале потока (для старых браузеров она обрабатывается как неизвестный им метод «*PRI*»). Зафиксируйте результат на снимке экрана.

2.2. Протокол HTTP2 содержит схожие с HTTP1.1 заголовки, но теперь их можно увидеть в читаемом виде только в декодере пакетов. Примените в качества фильтра отображения выражение «*http2*». Нажмите сочетание клавиш Ctrl+F. В открывшейся под фильтром строке поиска выберите тип поиска «*String*», область поиска «*Packet details*» и ведите в строку поиска комбинацию латинских символов, использованную в предыдущем пункте. Продолжайте поиск до тех пор, пока не найдёте пакет, содержащий одновременно искомый текст в декодере пакетов и HTTP2 метод «*POST*» (в столбце «*Info*» списка пакетов). Исследуйте в декодере заголовки найденного пакета. Какие из заголовков вам уже встречались при анализе сайта «max.zolotyh.su»? Перечислите их и зафиксируйте на снимке экрана некоторые из них (на ваш выбор).

2.3. Составьте фильтр отображения, который будет отображать только HTTP2 заголовки, содержащие методы «*POST*» (отправка данных) и «*GET*» (запрос страницы). Запишите этот фильтр. В столбце «*Info*» вы увидите много относительных URL: посмотрите весь этот список. Можете ли вы по содержащейся в них информации догадаться о роли хотя бы части из них в отображении страницы поиска? Если да, то выпишите примеры таких URL с комментариями.

2.4. В Wireshark есть большое количество инструментов, позволяющих извлекать статистику из захваченного трафика. Очень часто статистика оказывается даже важнее анализа отдельных пакетов. Воспользуемся одним из таки инструментов. Для этого примените фильтр отображения «*http2 || http*», а затем вызовите статистику по отправителям и получателям трафика при помощи меню «*Statistics → Endpoints*». В открывшемся окне перейдите на вкладку «*IPv4*», отметьте флажок «*Limit to display filter*» и отсортируйте список по убыванию количества байт, отправленных и принятых каждым узлом. Зафиксируйте полученный рейтинг на снимке экрана. Большая часть из этих узлов тоже участвовала в выдаче поисковой страницы. Это сервера, с которых подгружались картинки, подсказки, трекеры пользователя и другие ресурсы.

2.5. Откройте анализатор DNS пакетов в захваченном трафике: «*Statistics → Resolved Addresses*». Найдите имена узлов из рейтинга, полученного на предыдущем шаге. Запишите эти имена и соответствующие им IP-адреса в отчёт в табличной форме.

3. Дополнительная информация об IP-адресах

При анализе большого количества пакетов из разных источников не всегда бывает удобно оперировать только IP-адресами, так как они все похожи друг на друга и сливаются в восприятии пользователя. Wireshark позволяет заменить IP-адреса символьными (человекопонятными) именами, а так же дополнить их информацией о номере AS, владельце и географическом расположении. Эта информация содержится в специальных базах данных, не распространяемых вместе с Wireshark.

В анализаторе есть встроенная поддержка баз, предоставляемых одним из крупнейших поставщиков геолокации для IP – MaxMind. MaxMind предоставляет два варианта баз: бесплатная GeoLite и более полная коммерческая GeoIP. Для выполнения лабораторной работы будет достаточно первой из них, предоставляемой преподавателем. Самостоятельно загрузить актуальную версию базы можно на сайте **maxmind.com** после регистрации.

3.1. В папке «*%AppData%\Wireshark*» создайте файл *hosts* (обратите внимание, что его имя без расширения). Этот файл позволяет дополнить системный *hosts* без влияния на другие приложения. В созданный файл внесите строки, соответствующие вашему рабочему месту и рабочим места всех остальных бригад, выполняющих работу. Каждая строка должна иметь формат: «*ip_addres name*». Например, «*192.168.0.1 MyFavoriteGate*». Вместо *name* используйте псевдоним или имя пользователя, записанное латиницей без пробелов и специальных символов. Вы можете использовать одинаковый файл *hosts* для всей группы. Запишите содержимое файла в отчёт.

3.2. Перезапустите сетевой анализатор и запустите захват трафика. Выполните эхо-тестирование любого соседнего рабочего места, которое было записано в *hosts*. Остановите захват трафика. В меню «*View → Name Resolution*» выберите опцию «*Resolve Network Addresses*». Опишите словами, что изменилось в списке захваченных пакетов и покажите это изменение на снимке экрана.

3.3. Включите захват пакетов с фильтром, пропускающим только ARP-протокол. В течении 2-3 минут собирайте пакеты, затем остановите захват. Какие узлы проявляют активность в сети? Зафиксируйте результат на снимке экрана.

3.4. Подключите актуальные базы геолокации к анализатору. Для этого необходимо вызвать окно настроек при помощи пункта меню «*Edit → Preferences*». В открывшемся окне выбрать раздел «*Name Resolution*», в котором в пункте «*MaxMind database directories*» указать путь к каталогу с

файлами баз (рис. 0.1). Запустите захват трафика. В браузере откройте сайт **yandex.ru**. Остановите захват трафика. Найдите среди захваченных пакетов любой, отправленный к серверам Яндекса. В декодере пакета для протокола IP найдите информацию о геолокации адреса-получателя. К какой AS относится адрес и в каком городе расположен? Выделите строку с названием города и в контекстном меню для неё выберите пункт «*Apply as Column*». Зафиксируйте результат на снимке экрана.

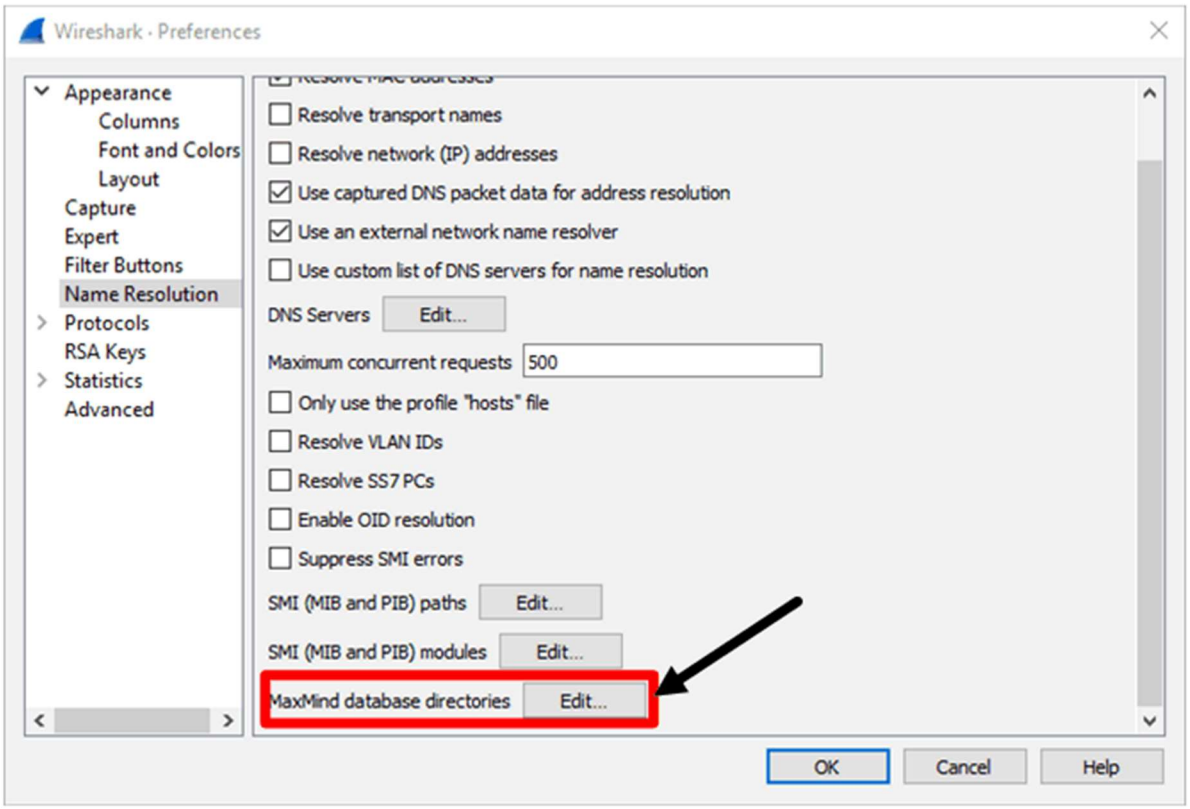

Рисунок 0.1. Подключение баз геолокации

3.5. Запишите фильтр отображения, который позволит выбрать только те пакеты, которые отправлены к IP-адресам, зарегистрированным за пределами России.

3.6. Удалите фильтр отображения и запустите захват пакетов. В течении нескольких минут посетите несколько веб-сайтов на ваш вкус. Остановите захват трафика. Вызовите уже знакомое вам по пункту 2.4 окно статистики по конечным точкам (*Endpoints*). Обратите внимание на заполненные столбцы *City*, «*AS Number*», «*AS Organization*». Включите разрешение имён (галочка «*Name resolution*»). Отсортируйте список по организации-владельцу AS. Зафиксируйте результат на снимке экрана.

3.7. При помощи кнопки *Map* в окне статистики отобразите расположение адресов на карте. Зафиксируйте результат на снимке экрана. Совпадают ли ваши ожидания о расположении адресов посещённых сайтов с полученной картой?

4. Другие источники трафика

Не всегда интересующий нас трафик может быть захвачен непосредственно в Wireshark. Иногда, например, для изучения незнакомого протокола, можно воспользоваться библиотеками заранее захваченных пакетов-примеров. Также трафик может быть записан непосредственно на маршрутизаторе или сервере при помощи популярной утилиты **tcpdump**[1] (Linux) или **pktmon** (Windows).

4.1. Познакомимся с IP-телефонией на готовом примере. Для перейдите в библиотеку образцов трафика **https://wiki.wireshark.org/SampleCaptures**[2]. Найдите там пример «*MagicJack+ short test call*»[3] (в разделе «*SIP and RTP*»), загрузите его на свой компьютер и откройте с помощью Wireshark. В анализаторе трафика. Используя приёмы, аналогичные ранее использованным для TCP, восстановите UDP потоки для SIP (передаёт информацию вызове) и RTP (передаёт голосовой трафик, в данном случае используя кодек G.711). Включите в отчёт снимки с экрана с их небольшими фрагментами.

4.2. Очистите фильтр отображения. Запустите анализатор звонков из меню «Telephony → *VoIP Calls*» и с его помощью прослушайте переданный разговор. Запишите в отчёт стенограмму этого разговора. Если у вас есть интерес к IP-телефонии, то рекомендуем изучить этот образец более подробно.

4.3. Создадим дамп трафика на серверной Linux машине. Для этого в VMWare Workstation запустите образ виртуальной машины, предоставленный преподавателем. Войдите в гостевую ОС от имения пользователя *student* с паролем «<*tpjgfcyjcnm&*»[4]. При помощи утилиты **ifconfig** определите IP-адрес виртуальной машины. Запишите его в отчёт.

4.4. Запустите захват сетевых пакетов с помощью команды «**sudo tcpdump -v <фильтр> -w ~/<имя>.cap**». Здесь **<фильтр>** – выражение в том же формате, что и фильтры захвата для Wireshark; **<имя>** – ваш псевдоним или номер рабочего места. В качестве фильтра используйте выражение, которое обеспечит захват только одноадресного трафика виртуальной машины (параметр *host* со значением IP-адреса, записанного на предыдущем шаге) и одновременно исключит из дампа ARP-протокол (параметр *arp*). Запишите в отчёт полученную команду.

4.5. Не прерывая захват пакетов в первой консоли, войдите во вторую консоль при помощи комбинации клавиш Alt+F2. В этой консоли выполните эхотестирование узла **yandex.ru**, трассировку пути к нему (в Linux команда

---

[1] Несмотря на своё название, утилита записывает любой трафик, а не только TCP.

[2] Существуют и другие библиотеки образцов трафика. Например, http://packetlife.net/captures

[3] MagicJack – это VoIP преобразователь, оснащённый разъёмом RJ-11 для подключения аналогового телефона и RJ-45 для подключения к Интернет через Ethernet.

[4] Несмотря на внешнюю сложность, это не очень хороший, словарный пароль. Не используйте его в реальных системах.

traceroute), а затем откройте заглавную страницу сайта в текстовом браузере **links**: «**links –anonymous http://ya.ru**» (зафиксируйте результат на снимке экрана). Вернитесь в первую консоль (Alt+F1) и остановите захват пакетов (Ctrl+C). Какое количество пакетов было захвачено?

4.6. Запустите приложение WinSCP, которое позволяет обмениваться с удалённой машиной файлами по протоколу SSH. Установите подключение по IP-адресу виртуальной машины с уже знакомыми вам именем пользователя и паролем. Загрузите созданный на предыдущем шаге дамп на свою основную машину и откройте его с помощью Wireshark. Зафиксируйте результат на снимке экрана.

4.7. Найдите в захваченном трафик эхо-запросы. Какую длину имеет поле данных и чем оно заполнено в Linux-утилите **ping**? Сравните с длиной и содержимым этого же поля в эхо-запросах, сформированных Windows-утилитой (при необходимости запустите ещё один экземпляр сетевого анализатора, а затем сформируйте и захватите необходимые пакеты).

4.8. Сравните IP-заголовки: какое значение *TTL* используется по умолчанию в пакетах, сформированных в Windows и Linux?

4.9. Исследуйте трафик утилиты **traceroute**. Какой зондирующий пакет и с какими параметрами она по умолчанию использует для «прощупывания» маршрута?

4.10. Найдите и восстановите TCP-поток, соответствующий запросу страницы **http://ya.ru**. Зафиксируйте результат на снимке экрана. Какой HTTP код пришёл в ответе сервера? Самостоятельно найдите значение этого кода и поясните: почему в данном случае ответ сервера не содержит данных, а только заголовки?

4.11. Не убирая фильтр TCP-потока, найдите в этом потоке первый пакет (тот, при помощи которого клиент инициирует соединение). Выпишите размер скользящего окна («*Windows size value*» и «*[Calculated window size]*[1]») и опции (с их значениями), установленные TCP-драйвером в Linux. Затем найдите второй пакет, отправленный клиентом. Как изменились в нём значения этих полей?

4.12. В ОС Windows запустите командную строку от имени администратора. Запустите захват пакетов встроенной утилитой: «**pktmon start –capture**», затем в браузере обратитесь к ресурсу «**ya.ru**», после чего отстановите захват командой «**pktmon stop**». Полученный файл преобразуйте в формат pacp: «**pktmon etl2pcap PktMon.etl –o PktMon.pcap**». Полученный файл откройте в новом окне Wireshark. Зафиксируйте результат на снимке экрана.

4.13. Для первого пакета TCP-соединения с сайтом «**ya.ru**» выпишите размер скользящего окна и опции, установленные TCP-драйвером Windows. Затем найдите второй пакет, отправленный клиентом. Как изменились в нём значения этих полей?

[1] Квадратные скобки в декодере пакетов указывают на значения, которые напрямую в пакете отсутствуют, но вычисляются на основе других полей.

4.14. Сравните поведение TCP-драйвера и трассировщика в Linux и Windows: может ли сервер различить подключения от этих ОС?