

Ettercap

Описание Ettercap

Ettercap — это всеобъемлющий набор для атаки "человек посередине" (MitM). Он умеет sniffить (прослушивать) живые соединения, фильтровать на лету содержимое передаваемых данных и многие другие трюки. Он поддерживает активное и пассивное вскрытие многих протоколов и включает многие функции для анализа сети и хостов.

Домашняя страница: <http://www.ettercap-project.org>

Авторы: Alberto Ornaghi (ALoR), Marco Valleri (NaGA), Emilio Escobar (exfil), Eric Milam (J0hnnnyBrav0), Gianfranco Costamagna (LocutusOfBorg)

Лицензия: GPLv2

Справка по Ettercap

Использование:

```
1 | ettercap [ОПЦИИ] [ЦЕЛЬ1] [ЦЕЛЬ2]
```

ЦЕЛЬ в формате MAC/IP/IPv6/ПОРТЫ (смотрите подробности ниже в [руководстве](#))

Далее mitm — это сокращение от Man in the middle, MitM-атака, т. е. атака Человек посередине.

```
1 | Оции sniffинга и атаки:
2 | -M, --mitm <METHOD:ARGS>    выполнить атаку mitm
3 | -o, --only-mitm              не sniffить, только выполнить атаку mitm
4 | -b, --broadcast              sniffить пакеты, предназначенные для трансляции
5 | -B, --bridge <IFACE>        использовать мостовой sniff (нужно 2 интерфейса)
6 | -p, --nopromisc              не переводить интерфейс в неразборчивый режим
7 | -S, --nosslmitm              не подделывать SSL сертификаты
8 | -u, --unoffensive            не перенаправлять пакеты
9 | -r, --read <file>           прочитать данные из pcapfile <файл>
10 | -f, --pcapfilter <строка>    установить <строку> pcap фильтром
11 | -R, --reversed               использовать обратное соответствие ЦЕЛИ
12 | -t, --proto <протокол>      sniffить только этот протокол (по умолчанию - все)
13 | --certificate <файл>        файл сертификата для использования с SSL MitM
14 | --private-key <файл>        файл личного ключа для использования с SSL MitM
15 |
16 | Тип пользовательского интерфейса:
17 | -T, --text                   использовать только текстовый интерфейс
18 | -q, --quiet                  не показывать содержимое пакетов
19 | -s, --script <CMD>          выполнить эти команды для GUI
20 | -C, --curses                 использовать интерфейс curses
21 | -D, --daemon                 демонизировать ettercap (нет интерфейса)
22 | -G, --gtk                    использовать интерфейс GTK+
23 |
24 | Оции ведения логов:
25 | -w, --write <файл>          записать перехваченные данные в pcapfile <файл>
26 | -L, --log <логфайл>         записать весь трафик в этот <логфайл>
27 | -l, --log-info <логфайл>    записать только пассивную информацию в этот <логфайл>
28 | -m, --log-msg <логфайл>     записать все сообщения в этот <логфайл>
29 | -c, --compress               использовать сжатие gzip для файлов логов
30 |
31 | Оции визуализации:
32 | -d, --dns                    резолвить ip адреса в адреса хостов
33 | -V, --visual <формат>       задать формат визуализации
34 | -e, --regex <регуляр>       визуализировать только пакеты, соответствующие этому
35 | -E, --ext-headers            записать расширенные заголовки для каждого rsk
36 | -Q, --superquiet             не показывать пользователей и пароли
37 |
38 | Оции LUA:
39 | --lua-script <скрипт1>,[<скрипт2>,...]    список скриптов LUA, разделённые
40 | --lua-args n1=v1,[n2=v2,...]              разделённые запятой список аргументов
41 |
42 | Общие оции:
43 | -i, --iface <iface>         использовать этот сетевой интерфейс
44 | -I, --liface                показать все сетевые интерфейсы
45 | -Y, --secondary <интерф>    список вторичных сетевых интерфейсов
46 | -n, --netmask <сетмаска>    установить эту <сетевую маску> на интерфейсе
47 | -A, --address <адрес>       установить этот локальный <адрес> для интерфейса
48 | -P, --plugin <плагин>       запустить этот <плагин>
49 | -F, --filter <файл>         загрузить <файл> фильтра (содержимое фильтра)
50 | -z, --silent                 не выполнять начальное ARP сканирование
51 | -6, --ip6scan                отправить ICMPv6 зонды для обнаружения IPv6 узлов на
52 | -j, --load-hosts <файл>     загрузить список хостов из <файла>
```

ПОИСК

ПОДПИСАТЬСЯ НА
ВЫХОД НОВЫХ СТАТЕЙ

Email*

SUBMIT

Онлайн книга

Аудит
безопасности Wi-
сетей
с Kali Linux

НОВЫЕ ПОСТУПЛЕНИЯ

- hcxtools
- hcxdumptool
- NMBscan
- NetBIOS Share Scanner
- NBTscan

РАЗДЕЛЫ

- Sniffing и Spoofing
- Анализ уязвимостей
- Анонимность
- Атаки на пароли
- Беспроводные атаки
- Веб приложения
- Инструменты по составлению отчётов
- Инструменты эксплуатации
- Криминалистические инструменты
- Обратная инженерия
- Поддержка доступа
- Сбор информации
- Стресс-тестирование
- Уязвимые среды и программы для тренировки

```
53 -k, --save-hosts <файл>      сохранить список хостов в <файл>
54 -W, --wifi-key <wkey>        использовать этот ключ для расшифровки Wi-Fi пакетов
55 -a, --config <конфиг>        использовать альтернативный файл <конфигурации>
56
57 Стандартные опции:
58 -v, --version                напечатать версию и выйти
59 -h, --help                  напечатать справку
```

Руководство по Ettercap

ИМЯ

ettercap — многоцелевой сниффер/фильтр контента для атак человек-посередине.

***** ВАЖНОЕ ЗАМЕЧАНИЕ *****

Начиная с ettercap NG (до 0.7.0) все опции были изменены. Изменена была даже спецификация целей. Пожалуйста, прочитайте внимательно это руководство.

СИНОПСИС

```
1 | ettercap [ОПЦИИ] [ЦЕЛЬ1] [ЦЕЛЬ2]
```

Если включён IPv6:

ЦЕЛЬ в виде MAC/IPs/IPv6/ПОРТЫ

В противном случае,

ЦЕЛЬ в виде MAC/IPs/ПОРТЫ

где IPs и ПОРТЫ могут быть диапазонами (к примеру, /192.168.0.1-30,40,50/20,22,25)

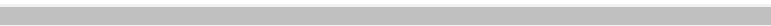
ОПИСАНИЕ

Ettercap был рождён как сниффер для коммутируемых локальных сетей (и, очевидно, даже для «захваченных»), но во время процесса разработки он получал больше и больше функций, которые изменили его в мощный и гибкий инструмент для атак человек-посередине. Он поддерживает активное и пассивное вскрытие многих протоколов (даже зашифрованных) и включает многие функции анализа сети и хостов (такие как отпечатки ОС).

Он имеет две главные опции сниффинга:

- UNIFIED (УНИФИЦИРОВАННЫЙ), этот метод сниффит все пакеты, которые проходят через кабель. Вы можете выбрать переводить или не переводить интерфейс в неразборчивый режим (опция **-p**). Пакеты не направленные на хост с запущенным ettercap будут автоматически перенаправляться используя уровень 3 роутинга. Поэтому вы можете использовать атаку mitm запущенную из различных инструментов и позволить ettercap модифицировать пакеты и перенаправлять их для вас.

Ettercap всегда отключает ip_forwarding ядра. Это сделано для предотвращения пересылки пакета дважды (от посредством ettercap и одним ядром). Это агрессивное поведение на шлюзе. Поэтому мы рекомендуем вам использовать ettercap ТОЛЬКО с ВКЛЮЧЁННЫМ БЕЗОБИДНЫМ РЕЖИМОМ

←  →

запуск его на шлюзе в наступательном режиме (offensive mode) не позволит пакетом быть перенаправленным обратно со вторичного интерфейса.

- BRIDGED (МОСТОВОМ), он использует два сетевых интерфейса и перенаправляет трафик с одного на другой при этом выполняя сниффинг и фильтрацию содержимого. Этот метод сниффинга совершенно незаметен, поскольку нет способа узнать, что кто-то посередине кабеля. Вы можете смотреть на этот метод как атака mitm на уровне 1. Вы будете посередине кабеля между двумя объектами. Не используйте его на шлюзах или он трансформирует ваш шлюз в мост. ПОДСКАЗКА: вы можете использовать движок фильтрации содержимого для отбрасывания пакетов, которые не должны пройти. Таким образом ettercap будет работать как встроенный IPS 😊
Вы можете выполнить атаки человек посередине во время использования унифицированного (unified) сниффинга и фильтровать процесс, т. е. вы можете запустить несколько атак одновременно или использовать ваш собственный инструмент для атаки. Ключевым моментом является то, что пакеты должны прибывать в ettercap с корректным mac адресом и отличным ip адресом (только такие пакеты будут перенаправлены).

Наиболее важными особенностями ettercap являются:

- **Поддержка SSH1:** вы можете сосниффить Пользователя и Пароль и даже данные SSH1 подключения. Ettercap является первой программой, способной сниффить SSH подключение в ПОЛНОМ ДУПЛЕКСЕ.
- **Поддержка SSL:** вы можете сниффить безопасные данные SSL... клиенту предоставляется фальшивый сертификат и сессия расшифровывается.
- **Вставка символов в установленном подключении:** вы можете вставлять символы для сервера (эмулирующие команды) или для клиента (эмулирующие ответы) поддерживающие соединение живым!!

- Шифрование данных антикриминалистика

НОВЫЕ ТЕМЫ НА ФОРУМЕ

- Произошла ошибка; возможно лента недоступна. Повторите попытку позже.

НОВЫЕ СООБЩЕНИЯ НА ФОРУМЕ

- Произошла ошибка; возможно лента недоступна. Повторите попытку позже.



НОВОСТИ ДРУЗЕЙ

- Ошибка «ModuleNotFoundError: No module named 'manipar» (РЕШЕНО)
- Ошибка «error: failed to commit transaction (conflicting files)» (РЕШЕНО)
- Ошибка «ERROR 1114 (HY000): line 19894: The table 'en' is full» (РЕШЕНО)
- LibreOffice перестал запускаться — как исправить? Сброс настроек LibreOffice при запуске (РЕШЕНО)
- Ошибка «PHP Fatal error: Uncaught mysqli_sql_exception: No database selected» (РЕШЕНО)

- **Фильтрация/отбрасывание пакетов:** Вы можете настроить скрипт фильтра для поиска конкретных строк (даже в шестнадцатеричном формате) в полезной нагрузке TCP или UDP и заменить их на ваши собственные или отбросить целый пакет. Движок фильтрации может искать на соответствие по любому полю сетевых протоколов и модифицировать любым образом на ваше усмотрение (смотрите [Etterfilter](#)).
- **Удалённый sniffing трафика через туннели и коверканье маршрута:** Вы можете поиграться с готовым интерфейсом Linux или использовать интегрированный плагин для sniffing туннелированных или с искажённым маршрутом соединениями и выполнить атаки mitm на них.
- **Поддержка плагинов:** Вы можете создавать ваши собственные плагины используя API ettercap'a.
- **Сбор паролей для:** TELNET, FTP, POP, RLOGIN, SSH1, ICQ, SMB, MySQL, HTTP, NNTP, X11, NAPSTER, IRC, RIP, BGP, SOCKS 5, IMAP 4, VNC, LDAP, NFS, SNMP, HALF LIFE, QUAKE 3, MSN, YMSG другие протоколы на подходе...).
- **Пассивное снятие отпечатков ОС:** вы пассивно сканируете локальную сеть (без отправки каких-либо пакетов) и собирать детальную информацию о хостах в LAN: операционную систему, запущенные службы, открытые порты, IP, mac адрес и производитель сетевого адаптера.
- **Убийство соединения:** из списка соединений вы можете убить соединения по вашему желанию.

ХАРАКТЕРИСТИКИ ЦЕЛЕЙ

Нет концепции ни ИСТОЧНИКА, ни НАЗНАЧЕНИЯ. Две цели предназначены для фильтрации проходящего трафика от одной к другой и наоборот (поскольку подключение является двунаправленным).

ЦЕЛЬ в формате MAC/IPs/ПОРТЫ.

ПРИМЕЧАНИЕ: Если включён IPv6, то ЦЕЛЬ в формате MAC/IPs/IPv6/ПОРТЫ.

Если хотите, вы можете пропустить любую из этих частей, это будет означать, что подходящим является ЛЮБОЕ значение

например

"//80" означает ЛЮБОЙ mac адрес, ЛЮБОЙ ip и ТОЛЬКО 80 порт

"/10.0.0.1/" означает ЛЮБОЙ mac адрес, ТОЛЬКО ip 10.0.0.1 и ЛЮБОЙ порт

MAC должен быть уникальным и иметь вид 00:11:22:33:44:55

IP — это диапазон IP в точечной нотации. Вы можете указать диапазон с помощью — (соединительной чёрточки, дефиса) и единичный ip с , (запятой). Вы также можете использовать ; (точки с запятой) для указания различных ip адресов.

например

"10.0.0.1-5;10.0.1.33" раскладывается в ip 10.0.0.1, 2, 3, 4, 5 и 10.0.1.33

ПОРТ — это диапазон ПОРТОВ. Вы можете задать диапазон с помощью - (дефиса) и единичные порты с помощью , (запятой).

например

"20-25,80,110" раскладывается в порты 20, 21, 22, 23, 24, 25, 80 и 110

ПРИМЕЧАНИЕ:

вы можете обратиться к соответствию ЦЕЛИ добавив в командную строку опцию **-R**. Т.е. если вы хотите sniffить ВСЕ трафик КРОМЕ одного входящего или исходящего на 10.0.0.1 вы можете задать `"/ettercap -R /10.0.0.1/"`

ПРИМЕЧАНИЕ:

ЦЕЛИ также ответственны за начальное сканирование локальной сети. Вы можете использовать их для ограничения сканирования только до подсети хостов в сетевой маске. Результат слияния двух целей будет просканирован. Помните, что неуказание цели означает «цели нет», но указание `"/"` означает «все хосты в подсети».

СБРОС ПРИВИЛЕГИЙ

ettercap нужны привилегии рута для открытия сокетов канального уровня. После фазы инициализации, привилегии рута больше не нужны, поэтому ettercap отбрасывает их на UID = 65535 (nobody — никто). Поскольку ettercap должен записывать (создавать) лог файлы, это должно осуществляться в директории с правами записи (к примеру, /tmp/). Если вы хотите сбросить привилегии до другого uid, вы можете экспортировать переменную окружения EC_UID со значением uid до которого вы хотите сбросить привилегии (например, `export EC_UID=500`) или установить корректный параметр в файле [etter.conf](#).

АТАКА SSL MITM

При выполнении атаки SSL mitm, ettercap заменяет реальный ssl сертификат на свой собственный. Фальшивый сертификат создаётся на лету и все поля заполняются в соответствии с реальным сертификатом, представленном на сервере. Изменяется только издатель и подписывается закрытым ключом содержащимся в файле 'etter.ssl.crt'. Если вы хотите использовать другой закрытый ключ, то вы должны сгенерировать этот файл. Для генерации файла cert используйте следующие команды:

```
1 openssl genrsa -out etter.ssl.crt 1024
2 openssl req -new -key etter.ssl.crt -out tmp.csr
3 openssl x509 -req -days 1825 -in tmp.csr -signkey etter.ssl.crt -out tmp.new
4 cat tmp.new >> etter.ssl.crt
5 rm -f tmp.new tmp.csr
```

ПРИМЕЧАНИЕ: SSL mitm is not available (for now) in bridged mode.

ПРИМЕЧАНИЕ: Вы можете использовать длинные опции --certificate/--private-key если вы хотите указать другой файл, а не etter.ssl.crt.

ОПЦИИ

Опции, которые имеют смысл вместе, как правило, могут быть объединены. Ettercap предупредит пользователя о неподдерживаемой комбинации опций.

ОПЦИИ СНИФФИНГА И АТАК

ettercap NG имеет новый унифицированный (unified) метод. Он предполагает, что ip_forwarding в ядре всегда отключён и перенаправление выполняет только ettercap. Каждый пакет с mac адресом пункта назначения равным mac адресу хоста и ip адресом пункта назначения отличным от привязанного к интерфейсу будет перенаправлен ettercap'ом. Перед их перенаправлением, ettercap может фильтровать контент, записывать или отбрасывать его. Не имеет значения, как эти пакеты перехвачены, ettercap обработает их. Вы даже можете использовать внешние программы для перехвата пакетов.

Вы имеете полный контроль над тем, что ettercap должен получать. Вы можете использовать внутренние атаки mitm, установить интерфейс в неразборчивый режим, использовать плагины или использовать любой метод, какой пожелаете.

ВАЖНОЕ ПРИМЕЧАНИЕ: если вы запустили ettercap на шлюз, не забудьте заново включить ip_forwarding после того, как вы убили ettercap. Поскольку ettercap отбрасывает свои привилегии, он не может для вас восстановить ip_forwarding.

-M, --mitm <МЕТОД:АРГУМЕНТЫ>

Атака MITM

Эта опция активирует атаку человек-посередине. Атака mitm совершенно независима от sniffingа. Цель этой атаки — перехватить пакеты и отправить их в ettercap. Движок sniffingа перенаправит их при необходимости.

Вы можете выбрать mitm атаку по своему предпочтению и также можете комбинировать некоторые из них для выполнения различных атак в одно время.

Если mitm метод требует какие-либо параметры, вы можете указать их после двоеточия. (например, -M dhcpr:ip_пул,маска сети,проч)

Доступны следующие mitm атаки:

arp ([remote],[oneway])

Этот метод осуществляет ARP отравление (пойзонинг — poisoning) атаки mitm. ARP запросы/ответы отправляются жертвам для отравления их ARP кэша. Когда кэш отравлен, жертвы будут отправлять все пакеты атакующему, который, в свою очередь, может их модифицировать и перенаправлять их реальным пунктам назначения.

В тихом режиме (опция **-z**) выбирается только первая цель, если вы хотите травить множество целей в тихом режиме используйте опцию **-j** для загрузки списка из файла.

Вы можете выбрать пустые цели и они будут расширены до «ЛЮБЫЕ» (все хосты в LAN). Список целей объединяется со списком хостов (созданным arp сканированием) и результат используется для выявления жертв атаки.

Параметр **"remote"** является опциональным и вы должны указать его если вы хотите sniffить удалённый ip адрес отравляя шлюз. В действительности если вы укажете жертву и шлюз в ЦЕЛЯХ, ettercap будет sniffить только соединение между ними, но чтобы ettercap был способен sniffить соединения, которые проходят через шлюз, вы должны использовать этот параметр.

Параметр "oneway" принудит ettercap отправлять только от ЦЕЛИ1 к ЦЕЛИ2. Полезно если вы хотите отравлять только клиента и не делать это с роутером (где может быть установлена программа вроде [arpwatch](#), следящая за изменениями арг).

Пример

цели: /10.0.0.1-5/ /10.0.0.15-20/

а список хостов: 10.0.0.1 10.0.0.3 10.0.0.16 10.0.0.18

ассоциации между жертвами будут:

1 и 16, 1 и 18, 3 и 16, 3 и 18

если цели перекрывают друг друга, ассоциации с идентичным IP адресом будут пропущены.

ПРИМЕЧАНИЕ: если вам удастся отравить клиента, то вы должны установить корректную таблицу в ядре, указав шлюз. Если ваша таблица маршрутизации некорректна, отравленный клиент не будет способен перемещаться по Интернету.

icmp (MAC/IP)

Эта атака выполняет ICMP перенаправление. Она отправляет фальсифицированное сообщение icmp редиректа хостам в локальной сети притворяясь лучшим маршрутом для Интернета. Все соединения в Интернет будут отправлены атакующему который, в свою очередь, перенаправит их реальному шлюзу. Результатом атаки будет ПОЛУДУПЛЕКС mitm. Будут отравлены только клиенты, поскольку шлюз не примет сообщения редиректа для непосредственно подключённой сети. УБЕДИТЕСЬ, ЧТО НЕ ИСПОЛЬЗУЕТЕ ФИЛЬТРЫ, КОТОРЫЕ МОДИФИЦИРУЮТ ДЛИНУ ПОЛЕЗНОЙ НАГРУЗКИ. Вы можете использовать фильтр для модификации пакетов, но длина должна быть той же самой, поскольку последовательности tcp не могут быть обновлены по обоим направлениям.

Вам нужно передать в качестве аргумента MAC и IP адрес реального шлюза для локальной сети.

Очевидно, вы должны быть способны sniffить трафик. Если вы на коммутаторе, вы должны использовать другую mitm атаку такую как отравление.

ПРИМЕЧАНИЕ: для ограничения редиректа заданной целью, укажите её как ЦЕЛЬ

Пример

-M icmp:00:11:22:33:44:55/10.0.0.1

будет перенаправлять все соединения, которые проходят через шлюз.

dhcр (ip_пул/сетевая маска/dns)

Эта атака реализует DHCP спуфинг (подмену). Она делает вид, что является DHCP сервером и пытается выиграть условия пути с реальным сервером, чтобы принудить клиента принять ответ атакующего. Этим манером ettercap способен манипулировать параметром шлюза и перехватывать весь исходящий трафик, генерируемый клиентами.

Результатом атаки будет ПОЛУДУПЛЕКС mitm. Поэтому убедитесь, что используете подходящие фильтры (смотрите выше секцию ICMP).

Вы должны передать пул IP для использования, сетевую маску и IP DNS сервера. Поскольку ettercap пытается победить в соревновании с реальным сервером, он НЕ ПРОВЕРЯТ был ли IP уже назначен. Вы должны указать пул СВОБОДНЫХ IP адресов для использования. Пул IP имеет тот же формат как и при задании цели.

Если клиент отправляет dhcр запрос (предполагая IP адрес) ettercap будет подтверждать получение этого IP и модифицировать только опцию шлюза. Если клиент делает исследование DHCP, ettercap будет использовать первый неиспользуемый IP адрес из указанного вами списка в командной строке. Каждое исследование потребляет один IP адрес. Когда список закончится, ettercap прекратит предлагать новые IP адреса и будет отвечать только на DHCP запросы.

Если вы не хотите предлагать какие-либо IP адреса, а только поменять информацию о роутере в DHCP запросах/подтверждения, вы можете указать пустой ip_пул.

БОЛЬШОЕ ПРЕДУПРЕЖДЕНИЕ: если вы укажете список IP, которые используются, вы наведёте беспорядок в вашей сети! В общем, используйте эту атаку с осторожностью. Она можете действительно натворить дел! Когда вы останавливаете эту атаку, жертвы всё ещё будут убеждены, что ettercap является шлюзом, пока не истечёт аренда адреса...

Пример:

```
-M dhcp:192.168.0.30,35,50-60/255.255.255.0/192.168.0.1
```

отвечать на DHCP оферты и запросы.

```
-M dhcp:/255.255.255.0/192.168.0.1
```

отвечать только на DHCP запросы.

port ([remote],[tree])

Эта атака реализует кражу портов. Эта техника полезна для сниффинга в коммутируемой среде, когда ARP отравление не эффективно (например, где используются статические прописанные ARP).

Она зафлуживает LAN (основываясь на опции `port_steal_delay` в `etter.conf`) пакетами ARP. Если вы не указали опцию **"tree"**, пунктом назначения MAC адреса каждого "украденного" пакета является тот же, что и у атакующего (другие NIC не будут видеть эти пакеты) MAC адрес источника будет одним из MAC адресов в списке хостов. Этот процесс «ворует» порт коммутатора каждого хоста жертвы в списке хостов. Используя низкую задержку, пакеты, предназначенные «украденным» MAC адресам, будут получены атакующим, выигравшим гонку условий с реальным владельцем порта. Когда атакующий получает пакеты «украденных» хостов, он останавливает процесс флудинга и выполняет ARP запросы к реальным пунктам назначения пакета. Когда он получает ARP ответ, то наверняка жертва получила «возврат» своего порта, поэтому ettercap может повторно отправлять пакеты по назначению как есть. Теперь мы можем заново начать процесс флудинга ожидая новых пакетов.

Если вы используете опцию **"tree"**, то пунктом назначения MAC адреса для каждого украденного пакета будет поддельный, поэтому эти пакеты будут распространяться на другие коммутаторы в дереве (если есть), но вы будете генерировать огромное количество трафика (в соответствии с `port_steal_delay`). Опция `"remote"` имеет то же значение что и в методе `"arp" mitm`.

Когда вы остановите эту атаку, ettercap отправит ARP запросы каждому украденному хосту для возвращения их портов коммутатора.

Вы можете выполнить как ПОЛУ, так и ПОЛНЫЙ ДУПЛЕКС `mitm` в соответствии с выбранной целью.

ПРИМЕЧАНИЕ: Используйте метод `mitm` только на коммутаторах локальной сети. Используйте его осторожно, он может приводить к понижению производительности или общему разрушению.

ПРИМЕЧАНИЕ: Вы НЕ можете использовать этот метод в режиме только-`mitm` (флаг **-o**), поскольку он использует движок сниффинга, и вы не можете использовать интерактивное внедрение данных.

ПРИМЕЧАНИЕ: Может быть опасным использование его совместно с другими методами `mitm`.

ПРИМЕЧАНИЕ: Этот метод `mitm` не работает на Solaris и Windows, из-за дизайна `ipcap` и `libnet` и отсутствия конкретной `ioctl()`.

Пример:

Целями являются: `/10.0.0.1/ /10.0.0.15/`

Вы будете перехватывать и визуализировать трафик между `10.0.0.1` и `10.0.0.15`, но вы будете получать весь трафик также для `10.0.0.1` и `10.0.0.15`.

Цель: `/10.0.0.1/`

Вы будете перехватывать и визуализировать трафик для `10.0.0.1`.

ndp ([remote],[oneway])

ПРИМЕЧАНИЕ: Этот MITM поддерживается только если включена поддержка IPv6.

Этот метод реализует атаку отравления [NDP](#), которая используется для MITM соединений IPv6. ND запросы/ответы отправляются жертвам для отравления их «соседского» кэша. Когда кэш отравлен, жертвы будут отправлять все IPv6 пакеты атакующему, который, в свою очередь, может модифицировать и перенаправлять их реальному пункту назначения.

В тихом режиме (опция **-z**) выбирается только первая цель, если вы хотите травить множество целей в тихом режиме, то используйте опцию **-j** для загрузки списка из файла.

Вы можете выбрать пустые цели и они будут расширены до «ЛЮБЫЕ» (все хосты в LAN). Список целей объединяется со списком хостов (созданным аргументом сканирования) и результат используется для выявления жертв атаки.

Параметр **"remote"** является опциональным и вы должны указать его если вы хотите sniffить удалённый ip адрес отравляя шлюз. В действительности если вы укажете жертву и шлюз в ЦЕЛЯХ, ettercap будет sniffить только соединение между ними, но чтобы ettercap был способен sniffить соединения, которые проходят через шлюз, вы должны использовать этот параметр.

Параметр **"oneway"** принудит ettercap отравлять только от ЦЕЛИ1 к ЦЕЛИ2. Полезно если вы хотите отравлять только клиента и не делать это с роутером (где может быть установлена программа вроде [arpwatch](#), следящая за изменениями аргумента).

Пример:

Цели: //fe80::260d:aff:fe6e:f378/ //2001:db8::2:1/

Диапазоны IPv6 адресов ещё не поддерживаются.

ПРИМЕЧАНИЕ: если вам удастся отравить клиента, то вы должны установить корректную таблицу в ядре, указав шлюз. Если ваша таблица маршрутизации некорректна, отравленный клиент не будет способен перемещаться по Интернету.

ПРИМЕЧАНИЕ: в IPv6 адрес локальной сети роутера обычно используются в качестве адреса шлюза. Следовательно, вам нужно установить локальный адрес роутера как одну цель и глобальный индивидуальный адрес жертвы как другую, чтобы настроить успешную атаку IPv6 MITM используя NDP травление.

-o, --only-mitm

Эта опция отключает процесс sniffинга и включает только атаку mitm. Полезно если вы хотите использовать ettercap для выполнения атак mitm и другой sniffер (такой как [wireshark](#)) для sniffинга трафика. Помните, что пакеты не перенаправляются ettercap'ом. Ядро будет ответственно за форвардинг. Не забудьте активировать функцию "ip forwarding" в вашем ядре.

-f, --pcapfilter <ФИЛЬТР>

Установить фильтр захвата из библиотеки pcap. Формат такой же как у tcpdump(1). Помните, что этого рода фильтры не sniffят пакеты с проводов, поэтому вы должны выполнить атаку mitm, ettercap не будет способен переправлять захваченные пакеты.

Эти фильтры полезны для уменьшения влияния сетевой нагрузки на модули декодирования ettercap.

-B, --bridge <IFACE>

Sniffинг МОСТОМ

Вам нужно два сетевых интерфейса. Ettercap будет перенаправлять с одного на другой весь трафик, который он видит. Это полезно для атаки человек посередине на физическом уровне. Она полностью незаметна, поскольку является пассивной и нет способа для пользователя увидеть атакующего.

Вы можете фильтровать контент во всём трафике как если бы вы были прозрачным прокси для «кабеля»,

ОФФЛАЙНОВЫЙ СНИФФИНГ

-r, --read <ФАЙЛ>

ОФФЛАЙНОВЫЙ sniffинг

С включённой этой опцией, ettercap будет sniffить пакеты из совместимого pcap файла вместо захвата из провода.

Это полезно, когда вы имеете файл дампа из tcpdump или [wireshark](#) и вы хотите сделать его анализ (поискать пароли или пассивные отпечатки).

Очевидно, во время sniffing файла вы не можете использовать «активный» sniffing (арг травление или соединение мостом) во время sniffing файла.

-w, --write <ФАЙЛ>

ЗАПИСАТЬ пакет в файл pcap

Это полезно если вы должны использовать «активный» sniffing (арг травление) на коммутируемой LAN, но вы хотите анализировать пакеты с помощью tcpdump или [wireshark](#). Вы можете использовать эту опцию для сдампливания пакетов в файл и затем загрузить его в ваше любимое приложение.

ПРИМЕЧАНИЕ: файл дампа собирает ВСЕ пакеты независимо от ЦЕЛИ. Это сделано от того, что вы можете захотеть записать даже протоколы, которые ettercap не поддерживает, поэтому вы можете анализировать их другими инструментами.

СОВЕТ: вы можете использовать опцию -w вместе с опцией -г. Таким путём вы будете способны фильтровать полезную нагрузку сдампленных пакетов или расшифровывать WiFi трафик с WEP-шифрованием и дампит их в другой файл.

ОПЦИИ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

-T, --text

Только текстовый интерфейс, только printf 😊

Это молчаливый интерфейс, нажимайте 'h' каждый раз для получения помощи о возможных действиях.

-q, --quiet

Тихий режим. Он может использоваться только в паре с консольным интерфейсом. Он не печатает текущие пакеты. Он полезен если вы хотите конвертировать pcap файл в лог файлы ettercap.

пример:

```
ettercap -Tq -L dumpfile -r pcapfile
```

-s, --script <КОМАНДЫ>

С этой опцией вы можете скармливать ettercap'у команды, как будто бы они были напечатаны на клавиатуре пользователем. Таким образом вы можете использовать ettercap внутри ваших любимых скриптов. Есть специальная команда, которую вы можете пропустить через эту команду: s(x). Эта команда приведёт к засыпанию на x секунд.

пример:

```
ettercap -T -s 'lq'
```

напечатает список хостов и выйдет

```
ettercap -T -s 's(300)olqq'
```

соберёт информацию за 5 минут, напечатает список локальных профилей и выйдет

-C, --curses

Графический интерфейс, основанный на Ncurses. Смотрите [ettercap_curses](#) для полного описания.

-G, --gtk

Славный GTK2 интерфейс (спасибо Daten...). Смотрите [ettercap-pkexec](#) для полного описания.

-D, --daemonize

Превратить ettercap в демон. Эта опция отсоединит ettercap от текущего контрольного терминала и установит его как демон. Вы можете комбинировать эту функцию с опцией "log" для записи трафика в фоне. Если демон вылетит по каким-либо причинам, он создаст файл ".ettercap_daemonized.log" в котором будет сообщена пойманная ettercap'ом ошибка. Более того, если вы хотите иметь полную отладку процесса демона, то рекомендуем вам перекомпилировать ettercap в режиме отладки.

ОБЩИЕ ОПЦИИ

-b, --broadcast

Говорит Ettercap обрабатывать пакеты приходящие с широковещательного адреса.

-i, --iface <ИНТЕРФЕЙС>

Использовать этот <ИНТЕРФЕЙС> вместо интерфейса по умолчанию. Этот интерфейс может быть ненастроен (требуется libnet >= 1.1.2), но вы в этом случае вы не можете использовать атаки MITM и вы должны установить флаг безобидного режима.

-I, --iflist

Эта опция напечатает список всех доступных сетевых интерфейсов, которые могут быть использованы с ettercap. Эта опция особенно полезна в Windows, где имя интерфейса не так очевидно, как в *nix.

-Y, --secondary <список интерфейсов>

Указывает список (или единичный) вторичных интерфейсов с которых захватывать пакеты.

-A, --address <АДРЕС>

Использовать этот <АДРЕС> вместо автоматически определённого для текущего интерфейса. Эта опция полезна если вы имеете интерфейс с множеством IP адресов.

-n, --netmask <МАСКА СЕТИ>

Использовать эту <МАСКУ СЕТИ> вместо маски сети ассоциированный с текущим интерфейсом. Эта опция полена, если у вас сетевая плата с ассоциированной маской сети класса B, а вы хотите сканировать (с арг сканированием) только класс C.

-R, --reversed

Меняет на противоположные соответствующие выбранные ЦЕЛИ. Это означает не(ЦЕЛИ). Всё, кроме выбранных ЦЕЛЕЙ.

-t, --proto <ПРОТОКОЛ>

Сниффить только пакеты ПРОТОКОЛА (по умолчанию это TCP + UDP).

Это полезно если вы хотите выбрать порт посредством указания ЦЕЛИ, но вы хотите дифференцировать tcp и udp.

ПРОТОКОЛ может быть "tcp", "udp" или "all" для обоих.

-b, --ip6scan

Отправлять ICMPv6 зонды для обнаружения ICMPv6 узлов на линии. Эта опция отправляет запросы пинга адресам всех-узлов, чтобы мотивировать активные IPv6 ответить. Вам не следует использовать эту опцию если вы пытаетесь скрыть себе. Следовательно, эта опция необязательная.

ПРИМЕЧАНИЕ: Эта опция доступна только если была включена поддержка IPv6.

-z, --silent

Не выполнять начальное ARP сканирование LAN.

ПРИМЕЧАНИЕ: у вас не будет списка хостов, следовательно, вы не сможете использовать функцию множественного отравления. Вы можете только выбрать два хоста для атаки ARP отравления, указав их в ЦЕЛЯХ.

-p, --nopromisc

Обычно ettercap переводит интерфейс в неразборчивый режим для сниффинга всего трафика на проводе. Если вы хотите сниффить только ваши подключения, используйте этот флаг для НЕ включения неразборчивого режима.

-S, --nossllitm

Обычно ettercap подделывает SSL сертификаты чтобы перехватить https трафик. Эта опция отключает это поведение

-u, --unoffensive

Каждый раз при запуске ettercap, от отключает IP форвардинг в ядре и начинает перенаправлять пакеты самостоятельно. Эта опция предотвращает это, поэтому ответственность за IP форвардинг остаётся на ядре.

Эта опция полезна если вы хотите запустить множество экземпляров ettercap. Вы будете иметь один экземпляр (один без опции **-u**) который будет перенаправлять пакеты, а другие экземпляры будут делать свою работу без перенаправления их. Иначе вы получите дубликаты пакетов.

Она также отключает внутреннее создание сессий для каждого подключения. Она увеличивает производительность, но вы будете неспособны модифицировать пакеты на лету.

Если вы хотите использовать атаку mitm, вы должны использовать различные экземпляры.

Вы должны использовать эту опцию интерфейс не настроен (без IP адреса.)

Это также полезно если вы хотите запустить ettercap на шлюз. Он не будет отключать форвардинг и шлюз будет корректно перенаправлять пакеты.

-j, --load-hosts <ИМЯ_ФАЙЛА>

Она может использоваться для загрузки списка хостов из файла, созданного с опцией **-k**. (смотри ниже)

-k, --save-hosts <ИМЯ_ФАЙЛА>

Сохраняет список хостов в файл. Полезна когда вы имеете много хостов и вы не хотите делать ARP шторм при запуске каждый раз, когда вы используете ettercap. Просто используйте эту опцию и соберите их все в файл, затем загрузить эту информацию из него используя опцию **-j** <имя_файла>.

-P, --plugin <ПЛАГИН>

Запустить выбранный ПЛАГИН. Многие плагины требуют указание цели, используйте ЦЕЛЬ как всегда. Используйте этот параметр несколько раз для выбора нескольких плагинов.

В консольном режиме (опция **-C**) отдельные плагины выполняются и затем приложение завершает свою работу. Активирован подхват плагинов и выполнение нормального сниффинга.

Чтобы получить список доступных внешних плагинов используйте "list" (без кавычек) как имя плагина (к примеру, ./ettercap -P list).

ПРИМЕЧАНИЕ: вы также можете активировать плагин непосредственно из интерфейсов (всегда нажимайте "h" для получения внутристрочковой помощи).

Больше подробностей о плагинах и о том, как их писать самому, смотрите на странице `man ettercap_plugin(8)`.

-F, --filter <ФАЙЛ>

Загрузить фильтр из файла <ФАЙЛ>. Фильтр должен быть скомпилирован с [etterfilter](#). Эта утилита скомпилирует скрипт фильтра и создаст совместимый с ettercap бинарный файл фильтра. Прочитайте [руководство etterfilter](#) для списка функций, которые вы можете использовать внутри скрипта фильтра. Любое количество фильтров может быть загружено указанием этой опции множество раз: пакеты проходят через каждый фильтр в том порядке, как они указаны в командной строке. Вы также можете загрузить скрипт без его включения, для этого добавьте к имени файла :0.

ПРИМЕЧАНИЕ: эти фильтры отличаются от тех, которые с --pcapfilter. Фильтр ettercap — это фильтр контента и может модифицировать полезную нагрузку пакета до его перенаправления. Фильтр Pcap используется для захвата только конкретных пакетов.

ПРИМЕЧАНИЕ: вы можете использовать фильтры на pcapfile для модификации его и сохранения в другой файл, но в этом случае вы должны обратить внимание что вы делаете, поскольку ettercap не будет повтор считать контрольные суммы, не будет расщеплять пакеты превышающие [MTU](#) (snaplen), вообще не будет ничего делать в этом роде.

-W, --wifi-key <KEY>

Вы можете указать ключ для расшифровки WiFi пакетов (WEP или WPA). Только успешно расшифрованные пакеты будут пропущены в декодерский стек, другие будут пропущены с выдачей сообщения.

Этот параметр имеет следующий синтаксис: type:bits:t:string. Где 'type' может быть: wep, wpa-pws или wpa-psk, 'bits' это битовая длина ключа (64, 128 или 256), 't' — это тип строки ('s' для строки и 'p' для пароля). 'string' строкой или экранированной шестнадцатеричной последовательностью.

пример:

```
--wifi-key wep:128:p:secret
```

```
--wifi-key wep:128:s:ettercapwep0
--wifi-key 'wep:64:s:\x01\x02\x03\x04\x05'
--wifi-key wpa:pwd:ettercapwpa:ssid
--wifi-key wpa:psk:
663eb260e87cf389c6bd7331b28d82f5203b0cae4e315f9cbb7602f3236708a6
```

-a, --config <КОНФИГ>

Загружает альтернативный файл конфигурации вместо дефолтного /etc/etter.conf. Это полезно, если вы имеете много предварительно настроенных файлов для различных ситуаций.

--certificate <ФАЙЛ>

Говорит Ettercap использовать указанный файл сертификата для атаки SSL MiTM.

--private-key <ФАЙЛ>

Говорит Ettercap использовать указанный файл частного ключа для атаки SSL MiTM.

ОПЦИИ ВИЗУАЛИЗАЦИИ

-e, --regex <РЕГУЛЯРНОЕ ВЫРАЖЕНИЕ>

Обрабатывать только пакеты, которые соответствуют этому регулярному выражению.

Эта опция полезна с **-L**. Она записывает только пакеты, которые соответствуют РЕГУЛЯРНОМУ ВЫРАЖЕНИЮ `posix`.

Она воздействует даже на визуализацию sniffленных пакетов. Она устанавливает показывать пакеты только соответствующие этому регулярному выражению.

-V, --visual <ФОРМАТ>

Используйте эту опцию для установления метода визуализации для отображаемых пакетов.

ФОРМАТ может быть следующим:

1	hex	Печатать пакеты в шестнадцатеричном формате.
2		
3		пример:
4		
5		строка "HTTP/1.1 304 Not Modified" становится:
6		
7		0000: 4854 5450 2f31 2e31 2033 3034 204e 6f74 HTTP/1.1 304 N
8		0010: 204d 6f64 6966 6965 64 Modified
9		
10	ascii	Печатать только "печатные" символы, другие отображаются как \
11		
12	text	Печатать только "печатные" символы, а остальные пропускать.
13		
14	ebcdic	Конвертировать текст EBCDIC в ASCII.
15		
16	html	Убрать все html тэги из текста. Тэгом является строка между <
17		
18		пример:
19		
20		<title>Это заголовок</title>, но последующая <строка> не буде
21		
22		Это заголовок, но последующая не будет отображена.
23		
24	utf8	Печатать пакеты в формате UTF-8. Используемая при преобразовани
25		кодировка устанавливается в файле etter.conf(5).

-d, --dns

Преобразовывать IP адреса в имена хостов.

ПРИМЕЧАНИЕ: это может серьёзно замедлить ettercap во время записи пассивной информации. Каждый раз при обнаружении нового хоста, будет выполнен запрос на dns. Ettercap хранит в кэше для уже преобразованных хостов для ускорения, но новые хосты требуют нового запроса и dns может понадобиться до 2 или 3 секунд для ответа по незнакомому хосту.

СОВЕТ: ettercap собирает dns ответы, он sniffит их в таблице преобразования, поэтому если вы указали не резолвить имена хостов, некоторые из них окажутся преобразованными, поскольку ответы были перехвачены ранее, воспринимайте это как бесплатное пассивное dns преобразование... 😊

-E, --ext-headers

Печатать расширенные заголовки для каждого отображаемого пакета. (например mac адреса)

-Q, --superquiet

Супер тихий режим. Не печатать собранных пользователей и пароли. Только сохранять их в профилях. Это может быть полезно для запуска ettercap только в текстовом режиме, но вы не хотите, чтобы зав зафлудили сообщениями диссекторов. Полезно при использовании плагинов, поскольку процесс сниффинга всегда активный, он будет печатать всю собранную информацию, с этой опцией вы можете подавить эти сообщения.

ПРИМЕЧАНИЕ: эта опция автоматически устанавливает опцию **-q**.

пример:

```
ettercap -TzQP finger /192.168.0.1/22
```

ОПЦИИ ЛОГИРОВАНИЯ

-L, --log <ФАЙЛ_ЛОГА>

Записывать все пакеты в бинарные файлы. Эти файлы могут быть пропарсены в [etterlog](#) для извлечения человечески читаемых данных. С этой опцией все перехваченные в ettercap пакеты будут записаны вместе со всей пассивной информацией (информация о хосте + пользователи и пароли) которую программа может собрать. По указанному ФАЙЛ_ЛОГА, ettercap создаст ФАЙЛ_ЛОГА.еср (для пакетов) и ФАЙЛ_ЛОГА.есі (для информации).

ПРИМЕЧАНИЕ: если в командной строке вы укажете эту опцию, вам не обязательно заботиться о привилегиях, поскольку файлы для записи открываются в фазе старта программы (с высокими привилегиями). Но если вы включили опцию логирования уже после старта ettercap, вы должны быть в директории, где может записывать uid = 65535 или uid = EC_UID.

ПРИМЕЧАНИЕ: лог файлы могут быть сжаты алгоритмом для уменьшения размеров данных опцией **-c**.

-I, --log-info <ФАЙЛ_ЛОГА>

Очень похоже на **-L**, но она записывает только пассивную информацию + пользователей и пароли для каждого хоста. Файл будет назван ФАЙЛ_ЛОГА.есі.

-m, --log-msg <ФАЙЛ_ЛОГА>

Она сохраняет в <ФАЙЛЕ_ЛОГА> все пользовательские сообщения, которые выводит ettercap. Это может быть полезно, когда вы используете ettercap в режиме демона или вы хотите отслеживать все сообщения. На самом деле, некоторые диссекторы печатают сообщения, но их информация нигде не сохраняется, поэтому это единственный способ ведение их учёта.

-c, --compress

Сжать файл лога алгоритмом gzip во время его дампа. [Etterlog](#) способен работать как с жатыми так и с несжатыми файлами логов.

-o, --only-local

Записывать информацию профилей принадлежащую только локальным хостам.

ПРИМЕЧАНИЕ: эта опция эффективна только в отношении профилей, собранных в памяти. Во время записи в файл ВСЕ хосты будут сохранены. Если вы хотите разбить их, используйте соответствующую опцию [etterlog](#).

-O, --only-remote

Сохраняет информацию о профилях принадлежащую удалённым хостам.

СТАНДАРТНЫЕ ОПЦИИ

-v, --version

Напечатать версию и выйти.

-h, --help

напечатать справку.

ПРИМЕРЫ

Здесь несколько примеров использования ettercap.

ettercap -Tr

Использовать консольный интерфейс и не переводить сетевой интерфейс в неразборчивый режим. Вы будете видеть только ваш трафик.

ettercap -Tzq

Использовать консольный интерфейс и не проводить ARP сканирования сети и быть тихим. Содержимое пакетов не будет отображено, но пользователи и пароли, а также другие сообщения, будут показаны.

ettercap -T -j /tmp/victims -M arp /10.0.0.1-7/ /10.0.0.10-20/

Загрузить список хостов из /tmp/victims и выполнить ARP атаку травления в отношении двух целей. Список будет объединён в цели и полученный список использован для ARP отравления.

ettercap -T -M arp // //

Выполнить атаку ARP сканирования в отношении всех хостов в локальной сети. БУДЬТЕ ОСТОРОЖНЫ!!

ettercap -T -M arp:remote /192.168.1.1/ /192.168.1.2-10/

Выполнить ARP отравление в отношении шлюза и хостов в локальной сети между 2 и 10. Опция 'remote' необходима для способности sniffить удалённый трафик, который хосты создают через шлюз.

ettercap -Tzq //110

Sniffить только протокол ror3 для каждого хоста.

ettercap -Tzq /10.0.0.1/21,22,23

Sniffить telnet, ftp и ssh подключения на 10.0.0.1.

ettercap -P list

Напечатать список доступных плагинов.

ФАЙЛЫ

~/.config/ettercap_gtk

Сохраняет постоянную информацию (например, размещение окна) между сессиями.

Примеры запуска Ettercap

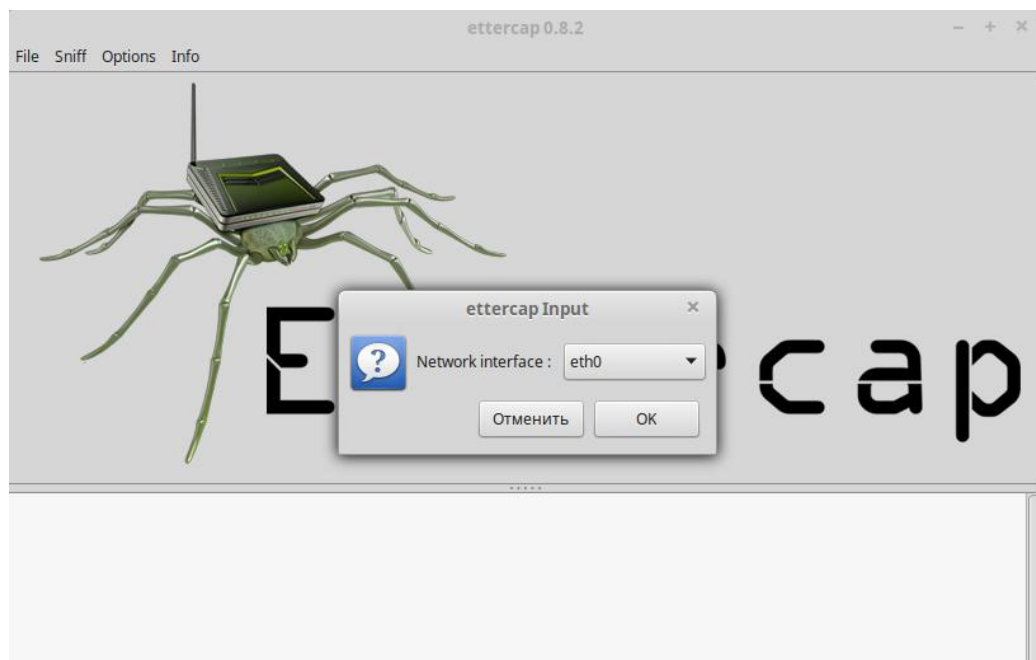
Переключите вашу машину в режим пересылки (форвардинга).

```
1 | echo "1" > /proc/sys/net/ipv4/ip_forward
```

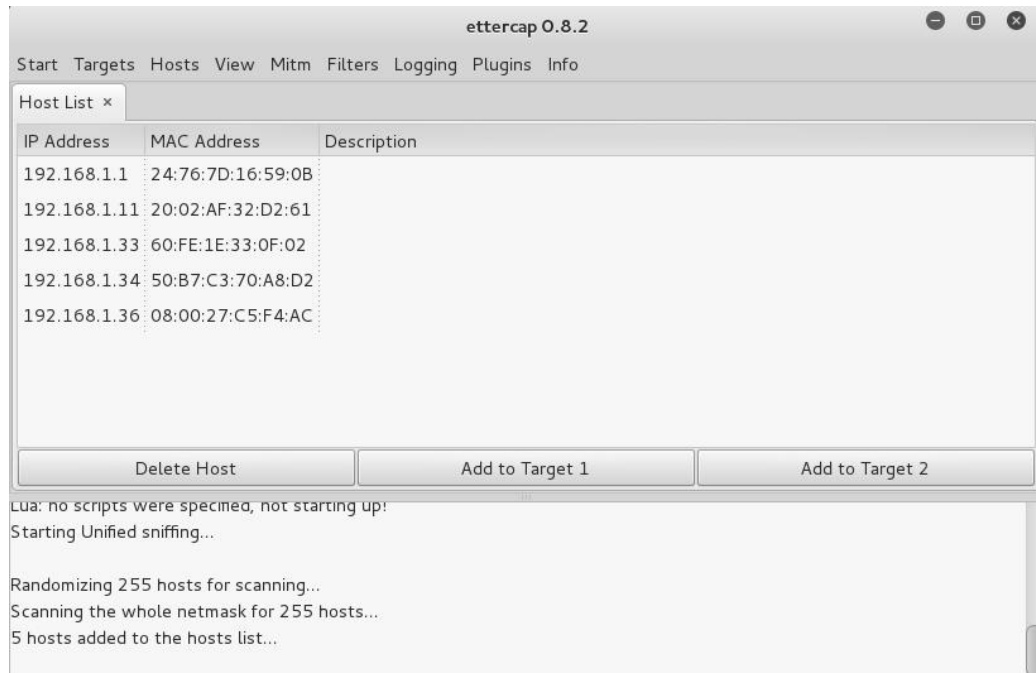
Запускаем графический интерфейс **(-G)**:

```
1 | sudo ettercap -G
```

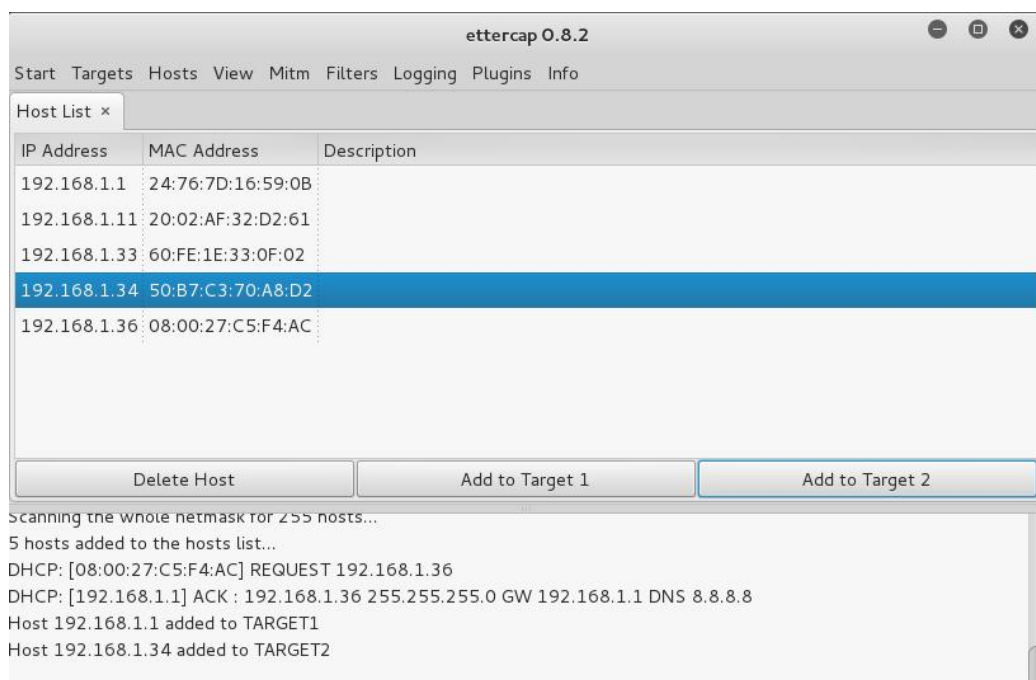
В меню выбираем **Sniff**, далее **Unified**, выбираем желаемый интерфейс:



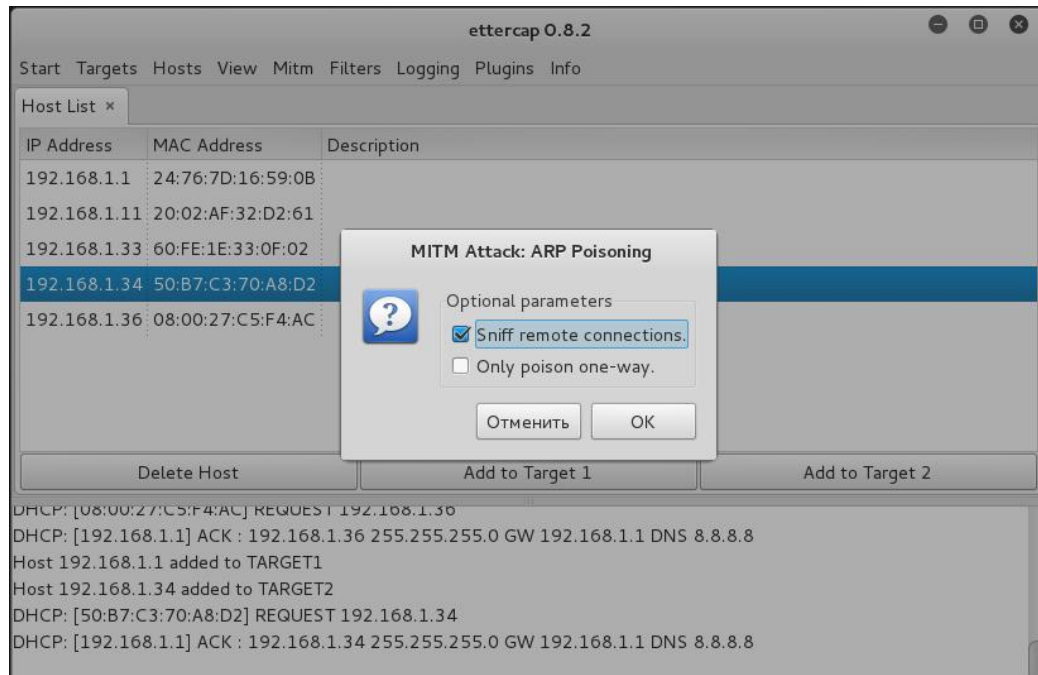
Теперь выбираем **Hosts**, в нём подпункт **Scan for hosts**. После окончания сканирования выберите **Hosts list**:



В качестве Цели1 выберите роутер (**Add to Target 1**), в качестве Цели2 выберите устройство, которое будете атаковать (**Add to Target 2**).



Теперь переходим к пункту меню **Mitm**. Там выберите **ARP poisoning...** Поставьте галочку на **Sniff remote connections**.



Теперь перейдите к меню **Sniff** и выберите там **Start sniffing**.

Чтобы убедиться, что sniffing работает, можно запустить urlsnarf, чтобы увидеть посещаемые адреса:

```
1 | urlsnarf -i eth0
```

или driftnet, чтобы увидеть открываемые на сайтах картинки:

```
1 | driftnet -i eth0
```

Установка Ettercap

Программа предустановлена в Kali Linux.

Установка Ettercap в Linux (на примере Debian, Mint, Ubuntu)

Установка из репозитория

```
1 | sudo apt-get install ettercap-common ettercap-graphical # только графический интерфейс
```

ИЛИ

```
1 | sudo apt-get install ettercap-common ettercap-text-only # только текстовый интерфейс
```

Запускать обязательно от имени администратора. К примеру так:

```
1 | sudo ettercap -G
```

Установка из исходных кодов (рекомендуется)

Установка зависимостей (на Kali Linux):

```
1 | sudo apt-get install git debhelper bison check cmake flex ghostscript libbsd-dev libncurses-dev
```

Установка зависимостей (на Linux Mint):

```
1 | sudo apt-get install git debhelper bison check cmake flex ghostscript libbsd-dev libncurses-dev
```

Скачиваем исходный код:

```
1 | git clone https://github.com/Ettercap/ettercap.git
```

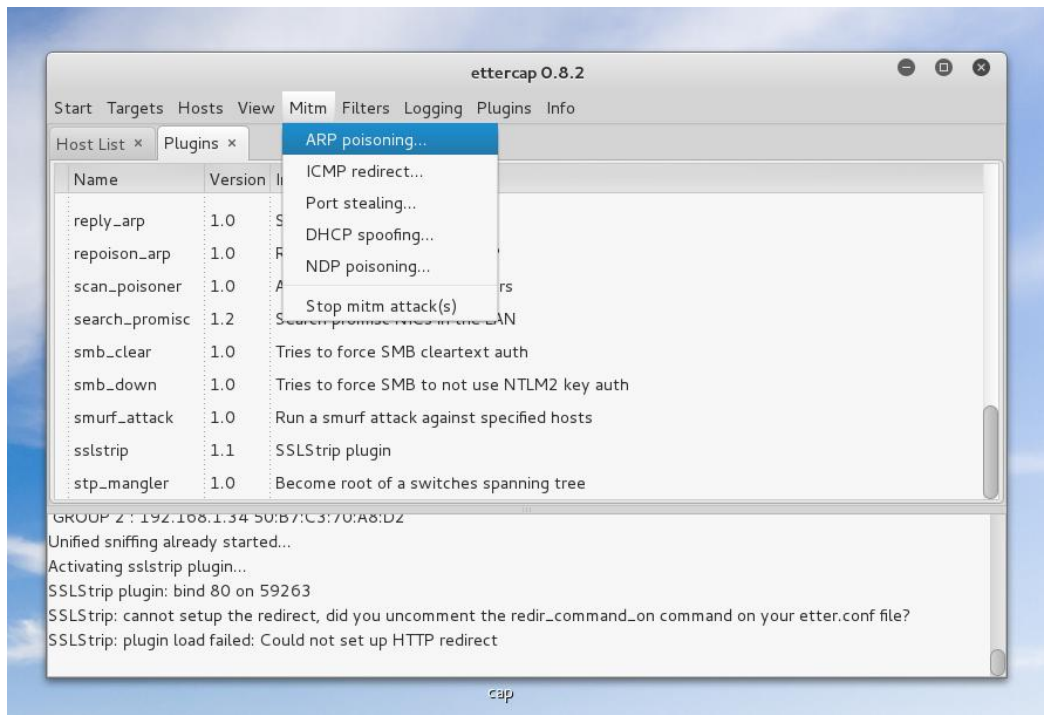
Собираем

```
1 | cd ettercap/
2 | mkdir build
3 | cd build
4 | cmake ENABLE_PDF_DOCS=On ../
5 | make
6 | sudo make install
```

Запускать обязательно от имени администратора. К примеру так:

```
1 | sudo ettercap -G
```

Скриншоты Ettercap



Инструкции по Ettercap

- [Инструкция по Ettercap: атака человек-посередине \(MitM\), перехват паролей, обход HSTS, подмена данных на лету, использование пользовательских фильтров и плагинов, подцепление на BeEF, заражение бэкдорами](#)
- [Видео «ARP-спуфинг в Kali Linux. Взлом, защита и описание технологии».](#)
- [Ettercap_curses и Ettercap-pxehex \(графические оболочки Ettercap\)](#)
- [etter.conf \(конфигурационный файл Ettercap\)](#)
- [ettercap-plugins \(плагины ettercap\)](#)
- [Etterfilter](#)
- [Etterlog](#)

Близкие программы:

- [MITMf](#) (82.6%)
- [Bettercap 1.6](#) (82.6%)
- [bettercap](#) (82.6%)
- [SSLsplit](#) (75%)
- [SSLstrip \(SSLStrip+\)](#) (69.3%)
- [Etterlog](#) (RANDOM - 59.3%)

Рекомендуется Вам:

COMMENTS ARE CLOSED

ПОИСК ПО САЙТУ