Metasploit общая практика

В 2003 году, хакеру, известному как «HD Moore», пришла идея разработать инструмент для быстрого написания эксплоитов. Так был рожден хорошо известный во всех кругах проект Metasploit. Первая версия фреймфорка была написана на языке Perl, содержавшая псевдографический интерфейс на базе библиотеки curses.

К 2007 году разработчики консолидировались, основав компанию Metasploit LLC; в это же время проект полностью переписали на Ruby и, частично на Си, Python и Ассемблер.

В октябре 2009 года, проект Metasploit был приобретен компанией Rapid7 с условием, что HD Moore останется техническим директором фреймворка, на что согласилась компания Rapid7.

Сегодня Metasploit является одной из популярнейших программ, имеющих самую большую базу эксплоитов, шеллкодов и кучу разнообразной документации, что не может не обрадовать.

Metasploit позволяет имитировать сетевую атаку и выявлять уязвимости системы, проверить эффективность работы IDS/IPS, или разрабатывать новые эксплоиты, с созданием подробного отчета. В народе его даже успели прозвать “хакерским швейцарским ножом”.

Благодаря переписанному, в основном на ruby, коду фреймворка, он остается кроссплатформенным, то есть не имеет конкретной привязки ни к какой ОС. HD Moore как-то продемонстрировал запуск Msfconsole на часах с linux прошивкой.

На сегодняшний день Metasploit содержится в нескольких linux-дистрибутивах:

— Kali linux (kali.org);
— Backtrack linux (backtrack-linux.org (поддержка прекращена));
— Pentoo (pentoo.ch);
— BlackArch (www.blackarch.org);
— Backbox (backbox.org).

С момента приобретения Фреймфорка, многое изменилось.
Например, появились PRO и Community версии, а в 2010 году, более упрощенная версия для «малоквалифицированных» пользователей — Metasploit Express.

Инструмент имеет несколько конфигураций:

1) Командная оболочка (msfconsole);
2) Веб-интерфейс (Metasploit Community, PRO и Express);
3) Графическая оболочка (Armitage, и более продвинутая версия — Cobalt strike).

Фреймворк: практика

Если у вас все еще нет Метасплоита — не беда, скачать его можно на официальном сайте metasploit.com.
Стоит отметить, что msf корректно работает только с базой PostgreSQL.

После того как мы загрузили пакет, откроем консоль и выполним следующее:

1) cd Download/ (перейдем в каталог загрузок);
2) chmod +X metasploit-latest-linux-installer.run (даем права на запуск);
3) sudo ./metasploit-latest-linux-installer.run (запускаем графическую установку).

(Более подробная документация по установке находится на сайте Metasploit).

В качестве краткого описания ознакомимся с основными понятиями, а также рассмотрим некоторые команды MSF.

Exploit — Фрагмент кода, использующий уязвимость в ПО или ОС для выполнения атаки на систему.
Module — Модуль, автоматизирующий процесс какой-либо атаки.
Shellcode — Шеллкод. Используется как полезная нагрузка эксплойта, обеспечивающая доступ к командной оболочке ОС.
Payload — Полезная, или смысловая нагрузка. Это код, который выполняется после успешного выполнения атаки. Видов нагрузки в msf немало.
«Stager» — Нагрузка, разбитая на части. Устанавливая соединение, шелл подгружается полностью.
«Reverse shell» — Бэкконнект шел.

«Meterpreter» — Пожалуй, один из популярных, если не самый популярный шелл. Имеет кучу возможностей: миграцию в процессы; XOR-шифрование, для обхода IDS и антивирусов; два вида dll-инжекции и т.д. Также можно выбрать «metsvc» нагрузку, что зальет и пропишет meterpreter как сервис. Более подробно о meterpreter можно почитать в статьях, ссылки на которые будут в конце статьи.

Закончим небольшой экскурс по meterpreter'у и перейдем к консоли msf.

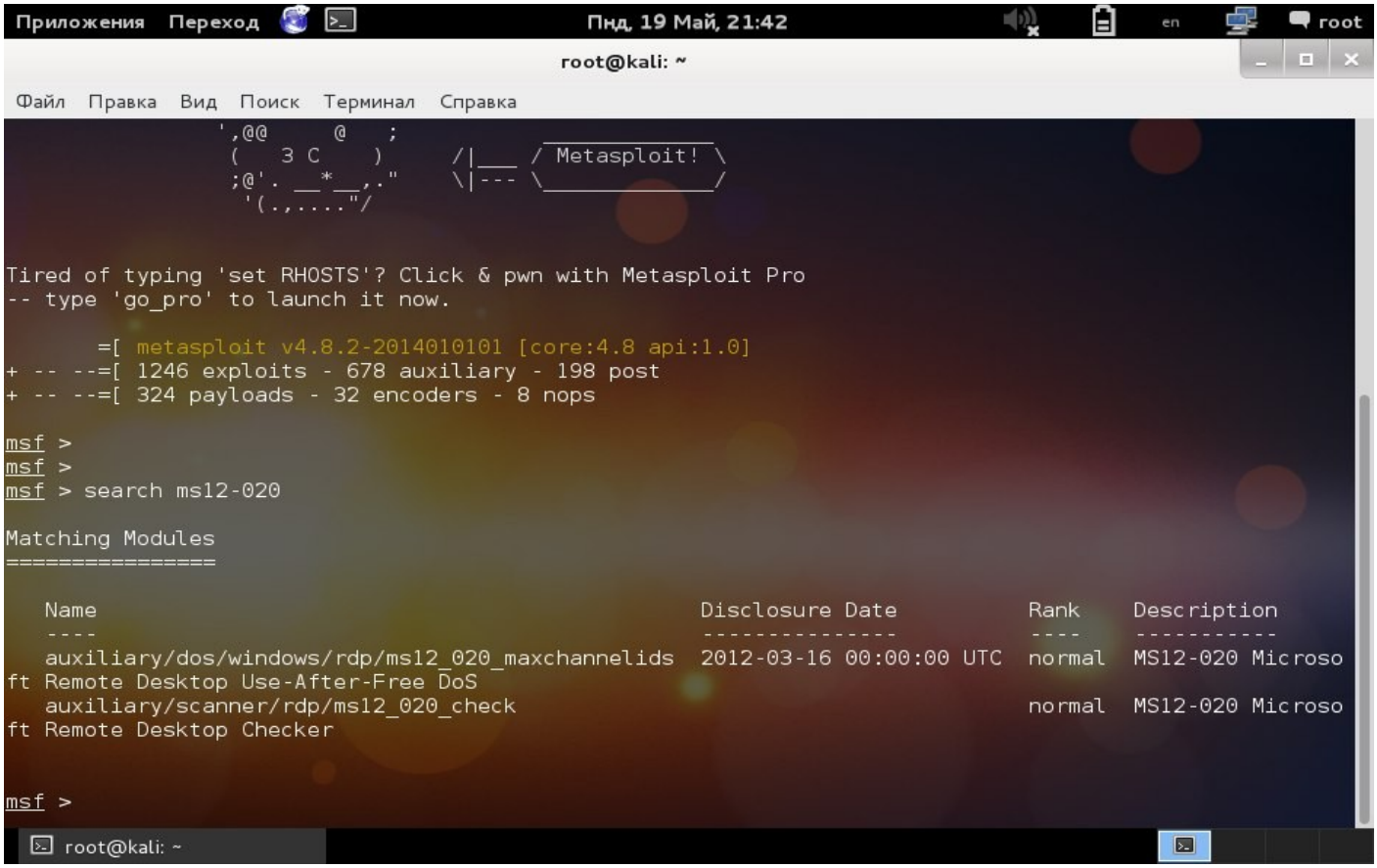
use — Выбор эксплоита
search — Поиск. Команда поиска более расширена; если вы забыли точное название или путь расположения эксплоита, она способна отобразить всю имеющуюся информацию
show options — Просмотр параметров для настройки. После выбора эксплоита, вы можете посмотреть какие опции доступны для настройки
show payload — Просмотр полезных нагрузок. Msf содержит множество полезных нагрузок; воспользовавшись этой командой можно также посмотреть рекомендуемые нагрузки для конкретного эскплоита или ОС
info — Просмотр подробной информации о полезной нагрузке
(info payload_name)
set — Установка параметров. Команда set устанавливает нужные параметры, например, RHOST(remote) и LHOST(local), или полезную нагрузку
(set PAYLOAD windows/shell/reverse_tcp)
check — Проверка хоста на уязвимость
exploit — Запуск сплоита. Когда цель выбрана и все возможное настроено, остается только завершающий этап — команда exploit

Также стоит отметить малоизвестную, но полезную фичу msf — возможность создания resource-скриптов. Сам resource-скрипт представляет из себя текстовый файл, содержащий последовательность команд для выполнения; также он позволяет выполнить ruby код.
Эти файлы очень удобны, и позволяют практически полностью автоматизировать без того легкий процесс тестирования. Например, это может пригодится для автоматического запуска сервера, либо очистки “мусора”.

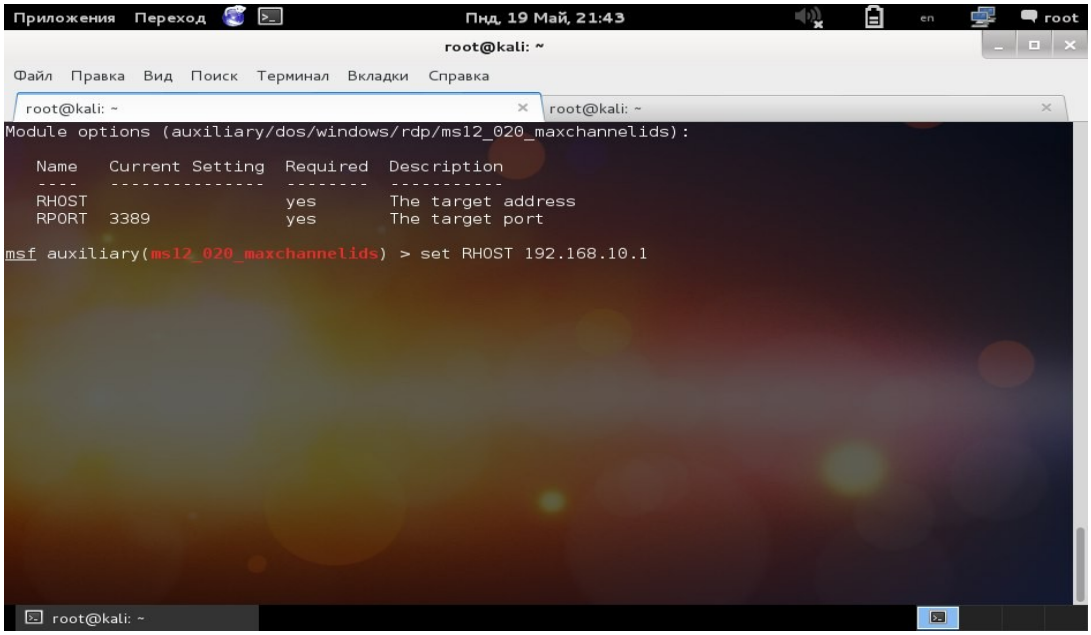
Заключение

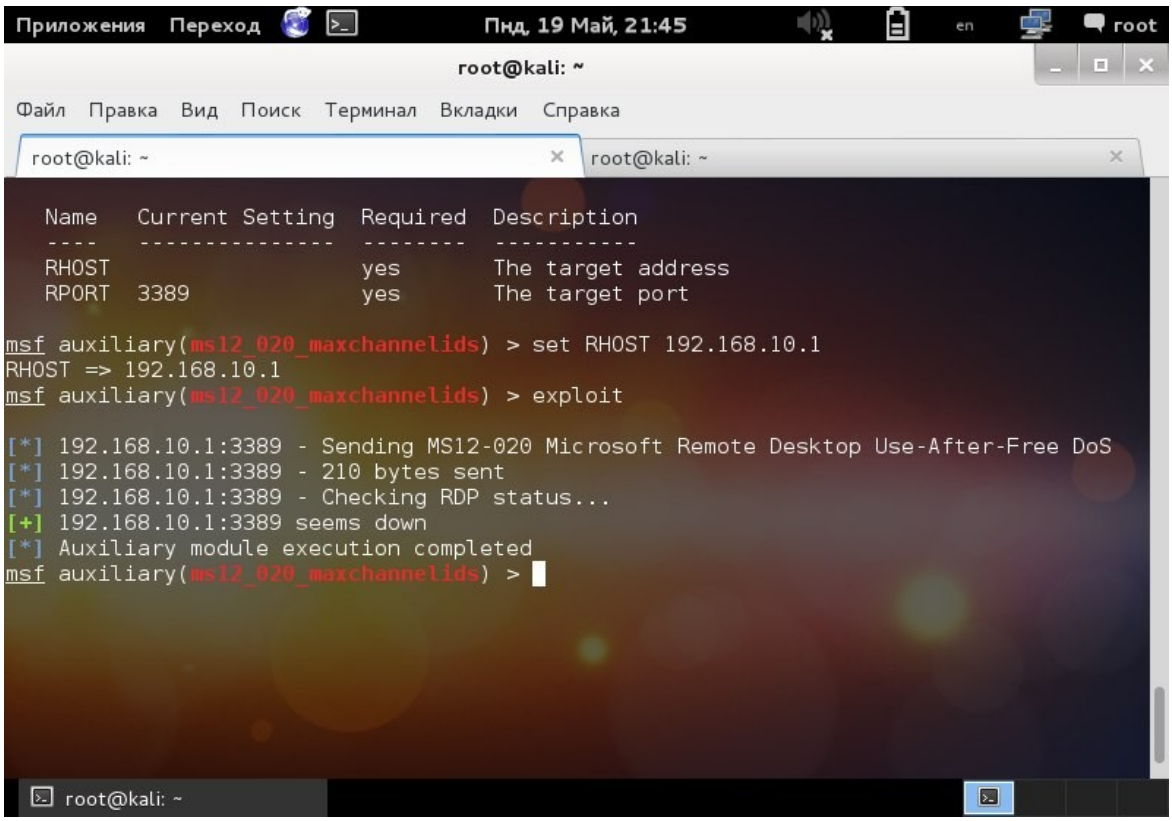
Уязвимость RDP протокола позволяет выполнит код на удаленной системе. Рассмотрим эксплоит ms12_020, приводящий к BSOD.

Сплоит располагается по адресу auxiliary/dos/windows/rdp/ms12_020_maxchennelids:



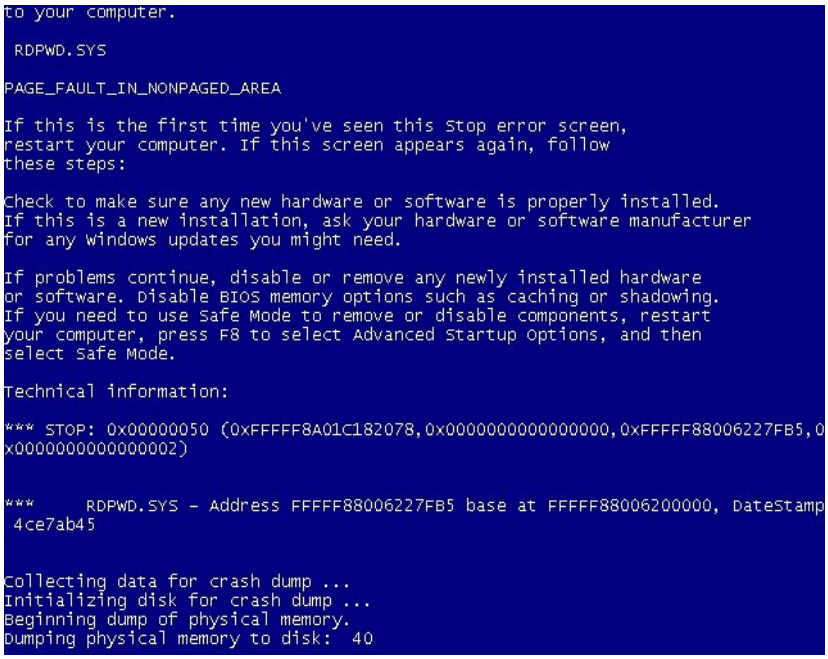
Выбрав сплоит, настроим его. Достаточно указать удаленный хост:





Как мы можем наблюдать на скриншоте выше, сессия установлена.

Правильно отработанный сплоит приводит к подобной картине:



По умолчанию, в нынешних версиях Windows RDP протокол не включен. Так что вам следует его предварительно включить