Лабораторная работа №1
по курсу «Методы защиты компьютерных сетей»

«Утилита netcat»

Цель работы: Изучить приемы применения утилиты netcat в разрезе информационной безопасности.

Введение

Netcat устанавливает и поддерживает TCP (Transmission Control Protocol) и UDP (User Datagram Protocol) соединения, читает и записывает данные по этим соединениям до тех пор, пока они не будут закрыты. Это основа работы сетевой подсистемы TCP/UDP, которая позволяет пользователям взаимодействовать по сети с помощью команд или скриптов с сетевыми приложениями и службами на прикладном уровне. Программа дает возможность увидеть пакеты TCP и UDP данных до того, как они будут упакованы в соответствии с протоколами более высокого уровня, такими как FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), или HTTP (Hypertext Transfer Protocol).

Примечание. Технически Netcat не устанавливает UDP-соединение, поскольку UDP-протокол работает и без него. В этой лекции, говоря о UDP-соединении, используемом Netcat, мы говорим об использовании Netcat в режиме UDP для того, чтобы начать передачу данных службе UDP, которая передает их принимающей стороне.

Netcat не делает ничего выдающегося. У этой программы нет красивого графического интерфейса пользователя (GUI), и она не выводит данные в виде красивого отчета. Но поскольку Netcat работает на таком базовом уровне, то он удобен для использования во многих ситуациях. Так как Netcat должен использоваться в паре с какими-либо еще программами или техническими приемами, неопытному пользователю его роль может показаться слишком преувеличенной. Управлять из командной строки файлом Readme новичку тоже довольно сложно. Тем не менее, к концу этой лекции вы увидите, что Netcat может стать одним из наиболее используемых инструментов в вашем арсенале.

Поскольку у Netcat так много способов применения, его часто сравнивают со "Швейцарским армейским ножом" для работы с TCP/IP и UDP протоколами.

Командная строка

Командная строка для Netcat выглядит как nc [options] host ports, где host - имя хоста или его IP-адрес для поиска, а ports - это или номер порта, или диапазон номеров портов (определяемый "m-n"), или несколько номеров портов разделенных пробелами.

Теперь вы полностью готовы посмотреть на те поразительные вещи, которые можно проделать с помощью Netcat. Внимательно присмотримся к каждой из опций командной строки, чтобы получить общие представления о возможностях.

- **-d**. Доступна только в Windows. Переводит Netcat в режим невидимки. Можно запустить программу в режиме прослушивания, не открывая окно режима MS-

DOS. Это также позволяет взломщикам лучше маскировать работающую программу от системных администраторов.

- **-e <command>**. Если Netcat скомпилирован с опцией GAPING_SECURITY_HOLE, программа может выполнять команду <command> всякий раз, когда кто-либо устанавливает соединение с прослушиваемым портом, до тех пор, пока клиент Netcat перенаправляет ввод/вывод работающей программе. Использовать эту опцию достаточно опасно, если вы не до конца представляете себе, что вы делаете. Это быстрый и простой способ открыть "черный ход" в вашу систему. Пример будет приведен далее.
- **-i <seconds>**. Интервал задержки между пересылками порций данных. Если через конвейер Netcat проходит файл, то программа ждет <seconds> секунд перед тем, как передать следующую строку, поступившую на вход. Если вы используете Netcat для управления несколькими портами на одном хосте, Netcat ждет <second> секунд перед тем, как соединится со следующим портом из перечисленных в строке. Это дает возможность немного замаскировать передачу данных или атаку системной службы, и это позволяет замаскировать сканирование портов от некоторых программных средств, анализирующих попытки внедрения, и от системных администраторов.
- **-g <route-list>**. Использование этой опции может быть весьма нетривиальным. Netcat поддерживает возможность маскировки начала маршрутизации (более подробно обсуждается в разделе "Создайте друга: подмена IP-адресов"). Вы можете определить до восьми -g опций в командной строке, чтобы заставить Netcat передавать трафик через определенные IP-адреса, которые обычно используются в случае, если вы подменяете исходящий IP-адрес, с которого поступает ваш трафик (например, для того, чтобы попытаться преодолеть брандмауэр или проверку разрешенных для доступа хостов). Используя этот прием на машине, с которой вы осуществляете управление процессом, вы можете заставить передаваемые пакеты возвращаться по указанному вами адресу вместо продвижения их по реальному направлению. Заметьте, что это обычно не срабатывает, поскольку большинство маршрутизаторов игнорируют опции источника маршрутизации, а многие фильтры и файрволы протоколируют такие попытки.
- **-G <hop pointer>**. Эта опция позволяет внести изменения в список маршрутизации, определенный параметром -g с тем, чтобы определить, к какому из адресов переходить. Поскольку IP-адрес - это четырехбайтовое число, этот аргумент всегда представляет собой число, кратное четырем, где 4 означает первый IP-адрес в списке, 8 - второй и так далее. Эта опция обычно используется в том случае, если вы пытаетесь так подделать список маршрутизации, чтобы он выглядел, как будто пакеты приходят откуда-то из другого места. Пропуская первые два IP-адреса, прописанные в списке, определенном опцией -g, и указав в параметре -G число 12, вы определите маршрутизацию пакетов непосредственно на третий адрес в вашем списке маршрутизации. Реальное содержание пакета по-прежнему будет содержать IP-адреса, которые были пропущены, создавая видимость, что пакеты пришли с одного адреса, тогда как на самом деле они пришли откуда-то еще. Этот прием позволяет скрыть, откуда вы пришли на хост, при использовании подмены адресов или списк а маршрутизации, но не факт, что у вас будет возможность получить ответ, поскольку он будет передаваться обратно по маршруту через ваши поддельные IP-адреса.
- **-l**. Эта опция переключает режим "прослушивания" Netcat. Она используется совместно с опцией -p, чтобы привязать Netcat к определенному TCP-порту и ожидать входящих соединений. Чтобы использовать UDP-порт, воспользуйтесь опцией -u.

- **-L**. Доступная только в Windows-версии программы, более жесткая опция режима "прослушивания", чем -l. Она указывает программе на необходимость перезапуска с теми же параметрами в случае, если соединение было закрыто. Это дает Netcat возможность отслеживать последующие соединения без вмешательства пользователя, каждый раз после завершения первоначального соединения. Как и в случае с опцией -l, эту опцию необходимо использовать совместно с опцией -p.
- **-n** сообщает Netcat, что не нужно осуществлять поиск каких-либо хостов. Если вы используете эту опцию, не следует указывать никаких имен хостов в качестве аргументов.
- **-o <hexfile>** обеспечивает создание шестнадцатеричного дампа данных и сохранение его в файле hexfile. Команда nc -o hexfile записывает данные, проходящие в обоих направлениях, и начинает каждую строку с символов < или > для обозначения соответственно входящих или исходящих данных. Чтобы записывать в файл только входящие данные, вам следует использовать команду nc -o <hexfile. Соответственно для записи только исходящих данных воспользуйтесь командой nc -o >hexfile.
- **-p <port>**. Опция позволяет вам определить локальный номер порта, который следует использовать Netcat. Этот аргумент требуется в случае, если вы используете опции -l или -L для режима прослушивания. Если эта опция не определена для исходящего соединения, Netcat будет использовать порт, который определен для этого в системе, что и делают большинство TCP или UDP клиентов. Имейте в виду, что в Unix-системах только пользователь root может определять номера портов меньше чем 1024.
- **-r**. Netcat выбирает локальный и удаленный порт случайным образом. Эта опция полезна в случае, когда Netcat используется для получения информации о большом интервале номеров портов в системе и при этом представить ситуацию так, чтобы это было в меньшей степени похоже на процедуру сканирования портов. В случае если эта возможность используется совместно с опцией -i и с достаточно большим интервалом, то велика вероятность, что сканирование портов не будет обнаружено без внимательного изучения системного журнала администратором.
- **-s** определяет исходящий IP-адрес, который Netcat использует для установки соединения. Эта опция позволяет взломщикам выполнять несколько изящных фокусов: скрыть свой IP-адрес или подделать что-либо еще. Но чтобы получить информацию, отправляемую на подмененный адрес, им необходимо использовать опцию определения порядка маршрутизации -g. Далее, используя режим прослушивания, вы можете многократно привязываться к уже прослушанному сервису. Все TCP- и UDP-сервисы работают с портами, но не каждый из них работает с конкретным IP-адресом. Многие службы по умолчанию прослушивают все доступные интерфейсы. Syslog, к примеру, прослушивает UDP-порт 514 для считывания трафика syslog. В то же время, если вы запустили Netcat на прослушивание 514 порта и использовали опцию -s для определения исходящего IP-адреса, любой трафик, проходящий через определенный вами IP-адрес, в первую очередь будет направляться через Netcat. Почему? Если сокет определяет и IP-адрес, и номер порта, это определяет его приоритет над сокетом, который не определяет обоих параметров. Позже мы расскажем об этом подробнее и продемонстрируем, как определить, какой сервис в системе может быть определен заранее.
- **-t**. Откомпилированный с опцией TELNET, Netcat может поддерживать взаимодействие с telnet-сервером в соответствии с установленными соглашениями, отвечая незначащей информацией, но дает вам возможность ввести информацию в ответ на приглашение ввести login, когда вы используете TCP-соединение по 23 порту.

- **-u**. Опция сообщает программе о необходимости использовать UDP-протокол вместо TCP, работая как в режиме прослушивания, так и в режиме клиента.
- **-v** определяет, насколько подробно программа информирует вас о том, что она делает. Если не использовать опцию -v, Netcat выдает только принятую информацию. Если опция -v использована один раз, вы сможете узнать, с каким адресом произошло соединение или какой адрес отслеживается в случае, если возникли какие-то проблемы. Повторное использование опции позволит узнать, какое количество данных было послано или принято до завершения соединения.
- **-w <seconds>** определяет промежуток времени, в течение которого Netcat ждет соединения. Этот параметр также сообщает, как долго следует ожидать после получения сигнала EOF (конец файла) на стандартный вход перед разрывом соединения и завершением работы. Это особенно важно, если вы посылаете команды удаленному серверу с использованием Netcat и ожидаете получить большой объем информации (к примеру, посылая веб-серверу HTTP команду на загрузку большого файла).
- **-z**. Если вы беспокоитесь только о том, чтобы определить, какой из портов открыт, вам следует использовать nmap. Но эта опция сообщает Netcat о необходимости послать достаточно данных для поиска открытых портов в заданном диапазоне значений.
  Более детальную информацию с примерами ищите в файле readme или в man-pages.

Задание для выполнения

Для выполнения заданий вам понадобится два компьютера: атакующий и жертва. Можете запускать программы как на одном, так и на другом.

1 Получите удаленный доступ к командной оболочке на машине с ОС Windows. Для этого запустите netcat на компьютере-жертве в режиме прослушивания входящих соединений, передавая вход-выход соединений командному интерпретатору.

С удаленной машины на жертве создайте и удалите пару каталогов, файлов.

Скройте запущенную программу netcat на жертве. Проверьте её наличие утилитой netstat и диспетчером процессов.

В чем достоинства и недостатки подобного метода взлома?

2 Выполните предыдущее задание с одним отличием: на машине-жертве должна работать ОС Linux.

3 Проведите сканирование диапазона портов 20-160 машины-жертвы. Включите режимы подробного информирования и посылки данных для того, чтобы иметь возможность просмотреть результат сканирования.

4 Передайте любой файл udp-соединением с одной машины на другую. Для этого вам придется вспомнить, что такое перенаправление ввода-вывода и как оно производится.

5 Сохраните вывод любого приложения жертвы в файл на другом компьютере. Попробуйте сделать это с помощью конвейера в Linux.

6 Попробуйте захватить любую службу на машине-жертве. Определите с помощью netstat, какие прослушиваются на машине порты, и запустите nc в режиме прослушивания одного из прослушиваемых портов. Установите соединение с другой машины, определите – какая программа выдает вам ответ: стандартная служба или ваш «захватчик».

Сохраняйте в текстовый файл все попытки соединения на указанную службу извне. Возможно, вам придется написать для этого небольшой shell-скрипт.

7 Получите http-заголовок и html-текст корневой страницы университетского веб-сайта.

Отчет

Отчет должен содержать все то, что вы делали, все, что вам выводилось в ответ, все ваши выводы, мысли и досужие домыслы. А также титульник, цель и вывод.