**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра САПР**

Методическое пособие

**«Сканирование сети при помощи Nmap»**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 9302 |  | Ширнин К.В. |
|  |  | Квитко Д.В. |
| Преподаватель |  | Горячев А.В. |

Санкт-Петербург

2022

**Оглавление**

[1. Установка Nmap 3](#_Toc117481297)

[1.1 Установка Nmap в Ubuntu и Debian 3](#_Toc117481298)

[1.2 Установка Nmap на CentOS и Fedora 3](#_Toc117481299)

[1.3 Установка Nmap на macOS 3](#_Toc117481300)

[1.4 Установка Nmap в Windows 4](#_Toc117481301)

[2. Синтаксис Nmap 5](#_Toc117481302)

[3. Справка по Nmap 7](#_Toc117481303)

# Установка Nmap

Nmap — это многоплатформенная программа, которую можно установить во всех основных операционных системах. Первоначально он был выпущен как инструмент только для Linux, а позже был перенесен на другие системы, такие как BSD, Windows и macOS.

Если вы предпочитаете графический интерфейс, а не командную строку, Nmap также имеет графический пользовательский интерфейс под названием Zenmap.

Официальные бинарные пакеты доступны для загрузки со страницы загрузки Nmap. Процедура установки проста и зависит от вашей операционной системы.

# Установка Nmap в Ubuntu и Debian

Nmap доступен из репозиториев Ubuntu и Debian по умолчанию. Чтобы установить его, запустите:

sudo apt update

sudo apt install nmap

# Установка Nmap на CentOS и Fedora

На CentOS и других производных от Red Hat запускаются:

sudo dnf install nmap

# Установка Nmap на macOS

Пользователи macOS могут установить Nmap, загрузив установочный пакет «.dmg» с сайта Nmap или через Homebrew:

brew install nmap

# Установка Nmap в Windows

Версия Nmap для Windows имеет некоторые ограничения и, как правило, немного медленнее, чем версия для UNIX.

Самый простой вариант установки Nmap в Windows — это загрузить и запустить самоустанавливающийся exe-файл.

Вы можете запустить Nmap в Windows либо из командной строки, либо запустив программу Zenmap. Для получения дополнительной информации о том, как использовать Nmap в Windows, ознакомьтесь с инструкциями по использованию после установки.

# Синтаксис Nmap

Команда запуска Nmap очень проста для этого достаточно передать ей в параметрах целевой IP адрес или сеть, а также указать опции при необходимости:

$ nmap опции адрес

Рассмотрим основные опции.

* **-sL** — просто создать список работающих хостов, но не сканировать порты nmap;
* **-sP** — только проверять доступен ли хост с помощью ping;
* **-PN** — считать все хосты доступными, даже если они не отвечают на ping;
* **-sS/sT/sA/sW/sM** — TCP сканирование;
* **-sU** — UDP сканирование nmap;
* **-sN/sF/sX** — TCP NULL и FIN сканирование;
* **-sC** — запускать скрипт по умолчанию;
* **-sI** — ленивое Indle сканирование;
* **-p** — указать диапазон портов для проверки;
* **-sV** — детальное исследование портов для определения версий служб;
* **-O** — определять операционную систему;
* **-T[0-5]** — скорость сканирования, чем больше, тем быстрее;
* **-D** — маскировать сканирование с помощью фиктивных IP;
* **-S** — изменить свой IP адрес на указанный;
* **-e** — использовать определенный интерфейс;
* **—spoof-mac** — установить свой MAC адрес;
* **-A** — определение операционной системы с помощью скриптов.

Теперь, когда мы рассмотрели все основные опции, давайте поговорим о том, как выполняется сканирование портов nmap.

# Справка по Nmap

nmap [Тип(ы) сканирования] [Опции] {определение цели}

ОПРЕДЕЛЕНИЕ ЦЕЛИ СКАНИРОВАНИЯ:

Можно использовать сетевые имена, IP адреса, сети и т.д.

  Пример: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254

**-iL <имя\_входного\_файла>**: Использовать список хостов/сетей из файла

**-iR <количество\_хостов>**: Выбрать произвольные цели

**--exclude <хост1[,хост2][,хост3],...>**: Исключить хосты/сети

**--excludefile <имя\_файла>**: Исключить список из файла

ОБНАРУЖЕНИЕ ХОСТОВ:

**-sL**: Сканирование с целью составления списка - просто составить список целей для сканирования

**-sn**: Пинг сканирование - отключить сканирование портов

**-Pn**: Рассматривать все хосты будто бы они онлайн -- пропустить обнаружение хостов

**-PS/PA/PU/PY[список\_портов]**: TCP SYN/ACK, UDP или SCTP обнаружение данных портов пингование заданных портов

**-PE/PP/PM**: Пингование с использованием ICMP эхо запросов, запросов временной метки и сетевой маски

**-PO**[список\_протоколов]: Пингование с использованием IP протокола

**-n/-R**: Никогда не производить DNS разрешение/Всегда производить разрешение [по умолчанию: иногда]

**--dns-servers <сервер1[,сервер2],...>**: Задать собственные DNS сервера

**--system-dns**: Использовать системный DNS преобразователь

**--traceroute**: Отслеживать путь к хосту

РАЗЛИЧНЫЕ ПРИЕМЫ СКАНИРОВАНИЯ:

**-sS/sT/sA/sW/sM**: TCP SYN/с использованием системного вызова Connect()/ACK/Window/Maimon сканирования

**-sU**: UDP сканирование

**-sN/sF/sX**: TCP Null, FIN и Xmas сканирования

**--scanflags <флаги>**: Задать собственные TCP флаги

**-sI <зомби\_хост[:порт]>**: "Ленивое" (Idle) сканирование

**-sY/sZ**: SCTP INIT/COOKIE-ECHO сканирования

**-sO**: Сканирование IP протокола

**-b <FTP ретранслирующий хост>**: FTP bounce сканирование

ОПРЕДЕЛЕНИЕ ПОРТОВ И ПОРЯДКА СКАНИРОВАНИЯ:

**-p <диапазон\_портов>**: Сканирование только определенных портов

    Пример: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9

**--exclude-ports <port ranges>**: Exclude the specified ports from scanning

**-F**: Быстрый режим - Сканировать меньше портов чем при сканировании по умолчанию

**-r**: Сканировать порты последовательно - не использовать случайный порядок портов

**--top-ports <количество\_портов>**: Сканировать <количество\_портов> наиболее распространенных портов

**--port-ratio <рейтинг>**: Сканировать порты с рейтингом большим чем <рейтинг>

ОПРЕДЕЛЕНИЕ СЛУЖБ И ИХ ВЕРСИЙ:

**-sV**: Исследовать открытые порты для определения информации о службе/версии

**--version-intensity <уровень>**: Устанавливать от 0 (легкое) до 9 (пробовать все запросы)

**--version-light**: Ограничиться наиболее легкими запросами (интенсивность 2)

**--version-all**: Использовать каждый единичный запрос (интенсивность 9)

**--version-trace**: Выводить подробную информацию о процессе сканирования (для отладки)

СКАНИРОВАНИЕ С ИПОЛЬЗОВАНИЕМ СКРИПТОВ:

**-sC**: эквивалентно опции --script=default

**--script=<Lua скрипты>**: <Lua скрипты> это разделенный запятыми список директорий, файлов скриптов или

           категорий скриптов

**--script-args=<имя1=значение1,[имя2=значение2,...]>**: Передача аргументов скриптам

**--script-args-file=имя\_файла**: передать скрипту NSE аргументы в файле

**--script-trace**: Выводить все полученные и отправленные данные

**--script-updatedb**: Обновить базу данных скриптов.

**--script-help=<Lua скрипты>**: Показать помощь о скриптах.

<Lua скрипты> разделённый запятой список файлов скриптов или

категорий скриптов.

ОПРЕДЕЛЕНИЕ ОС:

**-O**: Активировать функцию определения ОС

**--osscan-limit**: Использовать функцию определения ОС только для "перспективных" хостов

**--osscan-guess**: Угадать результаты определения ОС

ОПЦИИ УПРАВЛЕНИЯ ВРЕМЕНЕМ И ПРОИЗВОДИТЕЛЬНОСТЬЮ:

Опции, принимающие аргумент <время>, задаются в миллисекундах, пока вы не добавите 's' (секунды), 'm' (минуты),

  или 'h' (часы) к значению (напр. 30m).

**-T[0-5]**: Установить шаблон настроек управления временем (больше - быстрее)

**--min-hostgroup/max-hostgroup <кол\_хостов>**: Установить размер групп для параллельного сканирования

**--min-parallelism/max-parallelism <кол\_хостов>**: Регулирует распараллеливание запросов

**--min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <время>**: Регулирует время ожидания ответа на запрос

**--max-retries <количество\_попыток>**: Задает максимальное количество повторных передач запроса

**--host-timeout <время>**: Прекращает сканирование медленных целей

**--scan-delay/--max-scan-delay <время>**: Регулирует задержку между запросами

**--min-rate <число>**: Посылать запросы с интенсивностью не меньше чем <число> в секунду

**--max-rate <число>**: Посылать запросы с интенсивностью не больше чем <число> в секунду

ОБХОД ФАЙЕРВОЛОВ/IDS И СПУФИНГ:

**-f; --mtu <значение>**: Фрагментировать пакеты (опционально с заданным значениме MTU)

**-D <фикт\_хост1,фикт\_хост2[,ME],...>**: Маскировка сканирования с помощью фиктивных хостов

**-S <IP\_адрес>**: Изменить исходный адрес

**-e <интерфейс>**: Использовать конкретный интерфейс

**-g/--source-port <номер\_порта>**: Использовать заданный номер порта

**--proxies <url1,[url2],...>**: Переправлять подключения через прокси HTTP/SOCKS4

**--data <hex string>**: Append a custom payload to sent packets

**--data-string <string>**: Дописать пользовательскую ASCII строку к отправляемым пакетам

**--data-length <число>**: Добавить произвольные данные к посылаемым пакетам

**--ip-options <опции>**: Посылать пакет с заданным ip опциями

 --ttl <значение>: Установить IP поле time-to-live (время жизни)

**--spoof-mac <MAC\_адрес/префикс/название производителя>**: Задать собственный MAC адрес

**--badsum**: Посылать пакеты с фиктивными TCP/UDP контрольными суммами

ВЫВОД РЕЗУЛЬТАТОВ:

**-oN/-oX/-oS/-oG <файл>**: Выводить результаты нормального, XML, s|<rIpt kIddi3,

    и Grepable формата вывода, соответственно, в заданный файл

 -**oA <базовове\_имя\_файла>**: Использовать сразу три основных формата вывода

**-v**: Увеличить уровень вербальности (используйте -vv или более для усиления эффекта)

**-d**: Увеличить уровень отладки (используйте -dd или более для усиления эффекта)

**--reason**: Отобразить причину, по которой порт в конкретном состоянии

**--open**: Показывать только открытые (или возможно открытые) порты

**--packet-trace**: Отслеживание принятых и переданных пакетов

**--iflist**: Вывести список интерфейсов и роутеров (для отладки)

**--append-output**: Добавлять в конец, а не перезаписывать выходные файлы

**--resume <имя\_файла>**: Продолжить прерванное сканирование

**--stylesheet <путь/URL>**: Устанавливает XSL таблицу стилей для преобразования XML вывода в HTML

**--webxml**: Загружает таблицу стилей с Nmap.Org

**--no-stylesheet**: Убрать объявление XSL таблицы стилей из XML

РАЗЛИЧНЫЕ ОПЦИИ:

**-6**: Включить IPv6 сканирование

**-A**: Активировать функции определения ОС и версии, сканирование с использованием скриптов и трассировку

**--datadir <имя\_директории>**: Определяет место расположения файлов Nmap

**--send-eth/--send-ip**: Использовать сырой уровень ethernet/IP

**--privileged**: Подразумевать, что у пользователя есть все привилегии

**--unprivileged**: Подразумевать, что у пользователя нет привилегий для использования сырых сокетов

**-V**: Вывести номер версии

**-h**: Вывести эту страницу помощи.

ПРИМЕРЫ:

nmap -v -A scanme.nmap.org

nmap -v -sn 192.168.0.0/16 10.0.0.0/8

nmap -v -iR 10000 -Pn -p 80

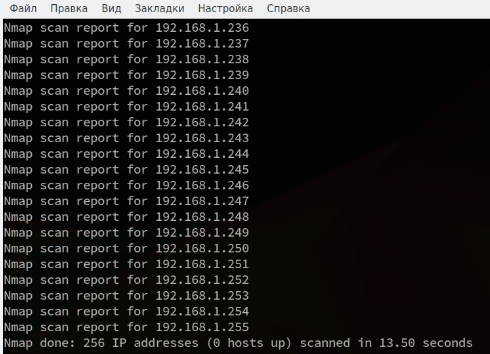
# Как пользоваться Nmap для сканирования портов в linux

Дальше рассмотрим примеры nmap. Сначала давайте рассмотрим как найти все подключенные к сети устройства, для этого достаточно использовать опцию -sL и указать маску нашей сети. В примере это 192.168.1.1/24. Маску вашей локальной сети вы можете найти, выполнив команду:

ip addr show

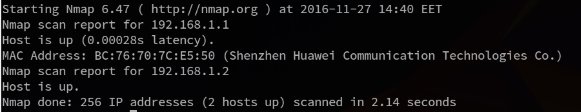
 Команда на сканирование сети nmap будет выглядеть вот так:

 nmap -sL 192.168.1.1/24



Иногда это сканирование может не дать никаких результатов, потому что некоторые операционные системы имеют защиту от сканирования портов. Но это можно обойти, просто использовав для сканирования ping всех ip адресов сети, для этого есть опция -sn:

 nmap -sn 192.168.1.1/24



Как видите, теперь программа обнаружила активные устройства в сети. Дальше мы можем сканировать порты nmap для нужного узла запустив утилиту без опций:

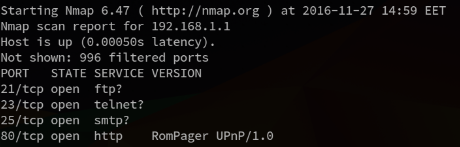
 sudo nmap 192.168.1.1



Теперь мы можем видеть, что у нас открыто несколько портов, все они используются каким-либо сервисом на целевой машине. Каждый из них может быть потенциально уязвимым, поэтому иметь много открытых портов на машине небезопасно.

Чтобы узнать более подробную информацию о машине и запущенных на ней сервисах вы можете использовать опцию -sV. Утилита подключится к каждому порту и определит всю доступную информацию:

sudo nmap -sV 192.168.1.1



На нашей машине запущен ftp, а поэтому мы можем попытаться рассмотреть эту службу подробнее с помощью стандартных скриптов nmap. Скрипты позволяют проверить порт более детально, найти возможные уязвимости. Для этого используйте опцию -sC и -p чтобы задать порт:

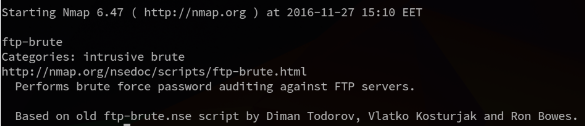
sudo nmap -sC 192.168.56.102 -p 21

Мы выполняли скрипт по умолчанию, но есть еще и другие скрипты, например, найти все скрипты для ftp вы можете командой:

sudo find /usr/share/nmap/scripts/ -name '\*.nse' | grep ftp

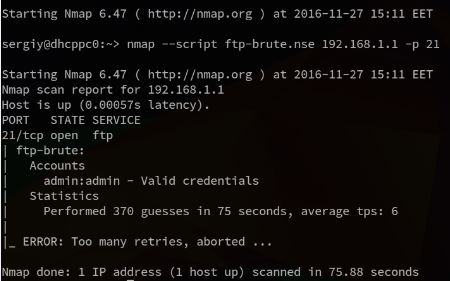
Затем попытаемся использовать один из них, для этого достаточно указать его с помощью опции —script. Но сначала вы можете посмотреть информацию о скрипте:

sudo nmap --script-help ftp-brute.nse



Этот скрипт будет пытаться определить логин и пароль от FTP на удаленном узле. Затем выполните скрипт:

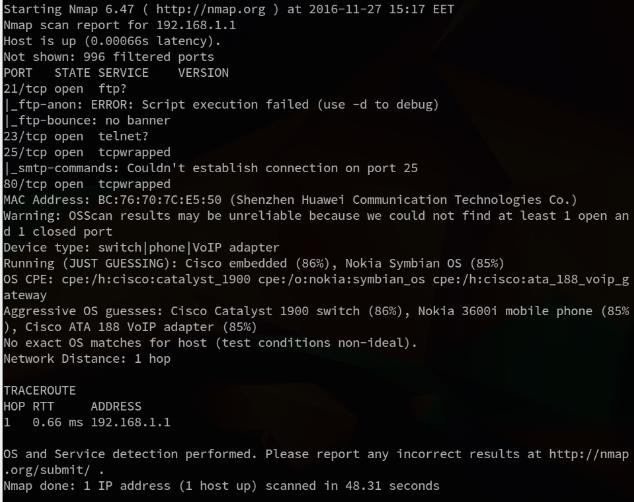
sudo nmap --script ftp-brute.nse 192.168.1.1 -p 21



В результате скрипт подобрал логин и пароль, admin/admin. Вот поэтому не нужно использовать параметры входа по умолчанию.

Также можно запустить утилиту с опцией -A, она активирует более агрессивный режим работы утилиты, с помощью которого вы получите большую часть информации одной командой:

sudo nmap -A 192.168.1.1



Обратите внимание, что здесь есть почти вся информация, которую мы уже видели раньше. Ее можно использовать чтобы увеличить защиту этой машины.