ГЛАВА

3

Сканирование сетевых хостов

Эта глава — ваш первый шаг на пути тестирования на проникновение. Независимо от того, профессионал вы или новичок, она поможет вам успешно провести сканирование сети. Вначале мы рассмотрим основы, которые вам необходимо знать, прежде чем начинать данную процедуру. Затем углубимся в подробности, чтобы увидеть, как сканировать сетевую цель.

В этой главе:

- основы построения сетей;
- определение живых хостов;
- сканирование портов;
- перечисление сервисов;
- отпечатки операционной системы;
- сценарии Nmap;
- сканирование поддоменов.

ОСНОВЫ ПОСТРОЕНИЯ СЕТЕЙ

Прежде чем вы начнете сканировать и идентифицировать хосты, вам необходимо понять основы работы в сети. Например, почему мы используем 10.0.0.1/16? Или что такое рукопожатие TCP? Так начнем же!

Сетевые протоколы

Ниже приведены два основных сетевых протокола, о которых необходимо знать для успешного сканирования сети.

TCP

Протокол управления передачей (Transmission Control Protocol, TCP) — основной протокол, используемый в сетевой инфраструктуре. Каждый сервер приложений (HTTP, FTP, SMTP и т. д.) задействует этот протокол для правильного соединения клиента с сервером.

TCP использует концепцию, называемую *трехсторонним рукопожатием*, для установления сетевого соединения. Чтобы начать сеанс TCP, клиент отправляет серверу SYN-пакет (синхронизируется). Сервер получает SYN и отвечает клиенту пакетом синхронизации/подтверждения (SYN/ACK). Наконец, клиент завершает диалог, отправляя пакет ACK серверу.

Например, на рис. 3.1 показан сценарий, в котором г-н Боб просматривает интернет и выполняет поиск в Google (на веб-сервере) с помощью своего браузера (клиента), посетив www.google.com.

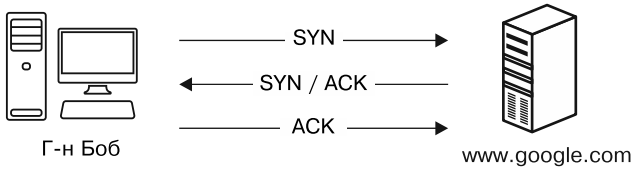

Рис. 3.1. TCP-рукопожатие

Важно понимать концепцию установления соединения TCP. Сетевые сканеры, такие как Nmap, используют ее для идентификации активных хостов, открытых портов и многого другого (вы узнаете больше об этом в следующих разделах).

Сетевой сниффер, такой как Wireshark, — хороший инструмент для изучения работы компьютерных сетей. Почему? Потому что сетевой сниффер будет прослушивать весь входящий и исходящий трафик через сетевую карту.

Чтобы запустить Wireshark, просто введите его имя (`$ wirehark`) в окне терминала. Далее вам нужно будет выбрать сетевой интерфейс; это либо Ethernet `eth0`, либо Wi-Fi `wlan0`. В данном случае мы используем `eth0`, а затем нажимаем кнопку Start в верхнем левом углу экрана (рис. 3.2).

Скриншот, показанный на рис. 3.3, снят из Wireshark на моей Kali (10.0.0.20), когда я открыл свой браузер и перешел на Google.com. Если присмотреться, то можно увидеть пакеты [SYN], [SYN ACK] и [ACK].

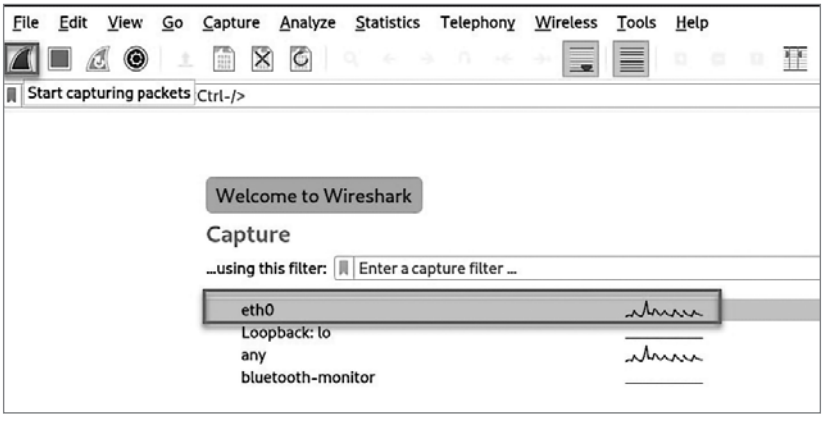

Рис. 3.2. Выбор сетевого интерфейса Wireshark

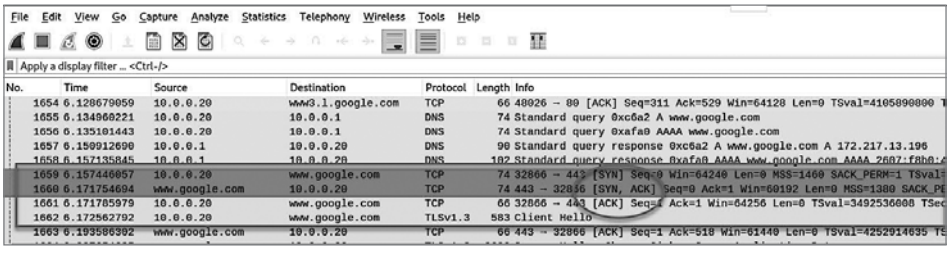

Рис. 3.3. Захват трафика с Wireshark

UDP

Протокол пользовательских дейтаграмм (User Datagram Protocol, UDP) представляет собой *сетевое соединение без установления соединения*. В отличие от TCP-соединения, UDP-клиент и сервер не гарантируют передачу пакетов, поэтому в UDP нет трехстороннего рукопожатия. Примерами приложений, использующих UDP, являются потоковое аудио и видео — вам нужна производительность при таких соединениях. Далее в этой главе в табл. 3.3 показаны наиболее популярные приложения с соответствующими протоколами TCP или UDP.

Другие сетевые протоколы

TCP и UPD — самые популярные сетевые протоколы, но существуют и другие типы протоколов. Сейчас мы рассмотрим их.

ICMP

ICMP (Internet Control Message Protocol) служит для проверки возможности подключения. С его помощью инструмент Ping проверяет, работает ли хост в сети (`traceroute` также использует его по умолчанию). В следующем примере мы будем пинговать IP-адрес 10.0.20.1 и проверять соединение ICMP в Wireshark:

```
root@kali:~# ping 10.0.0.1 -c 3
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.706 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.725 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.506 ms

--- 10.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2033ms
rtt min/avg/max/mdev = 0.506/0.645/0.725/0.099 ms
```

На рис. 3.4 показан Wireshark с фильтрацией по `icmp`:

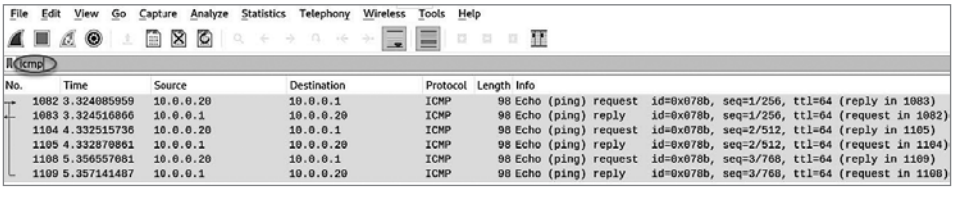

Рис. 3.4. Фильтр Wireshark ICMP

ARP

Протокол разрешения адресов (Address Resolution Protocol, ARP) — это механизм, который сопоставляет адреса IPv4 с MAC-адресами. Данная концепция важна для работы внутренней сети. Маршрутизаторы соединяются друг с другом через интернет с помощью IP-адресов (уровень 3), но как только пакет попадает в вашу сеть, будет взята таблица ARP с использованием MAC-адресов (уровень 2). Вы можете применить команду `arp -a`, чтобы получить список элементов внутри таблицы ARP (сохраненной локально на вашем хосте):

```
root@kali:~# arp -a
? (10.0.0.10) at 70:5a:0f:f6:fc:3a [ether] on eth0
USGPRO (10.0.0.1) at b4:fb:e4:2f:04:3d [ether] on eth0
```

Под уровнями понимаются уровни OSI, показанные в табл. 3.1. Модель OSI (Open Systems Interconnection) разделяет сетевое соединение на разные уровни.

Таблица 3.1. Уровни OSI

НОМЕР	НАИМЕНОВАНИЕ	ПРИМЕРЫ ПРОТОКОЛОВ	ПРИМЕРЫ УСТРОЙСТВ
1	Физический	Ethernet и пр.	Кабели
2	Канальный	MAC, VLAN и пр.	Свитчи
3	Сетевой	IPv4/v6 ICMP и пр.	Роутеры
4	Транспортный	TCP UDP	NA
5	Сеансовый	NA	NA
6	Уровень представления	NA	NA
7	Прикладной	FTP, HTTP, Telnet и пр.	Сетевые экраны, прокси и т. д.

IP-адресация

Интернет-протокол (Internet Protocol, IP) — один из основных столпов сети, позволяющий компьютерам взаимодействовать друг с другом. IP-адреса делятся на две версии: IPv4 и IPv6.

IPv4

IPv4 является 32-битным, но всегда представлен в десятичном формате, например 192.168.0.1, что равно 11000000.10101000.00000000.00000001. Его проще записать в десятичном формате, а не в двоичном, не так ли?

IP-адреса делятся на общедоступные (используются в интернете) и частные (используются в локальной сети). Ваш общедоступный IP-адрес, вероятно, предоставляется автоматически интернет-провайдером (ISP), если вы не купили статический общедоступный IP-адрес.

Ниже представлены диапазоны частных IPv4-адресов:

- от 10.0.0.0 до 10.255.255.255 (10.x.x.x), примерно с 16 миллионами адресов;
- от 172.16.0.0 до 172.31.255.255 (от 172.16.x.x до 172.31.x.x), примерно с 1 миллионом адресов;
- от 192.168.0.0 до 192.168.255.255 (192.168.x.x), около 65 тысяч адресов.

Подсети и CIDR

Роль подсети — разделить сеть на более мелкие диапазоны (сегментация сети). Подсеть определяет количество хостов внутри диапазона IP-адресов. Например,

192.168.0.1 может иметь маску подсети 255.255.255.0, что говорит о нашей возможности использовать 254 хоста внутри этого диапазона IP-адресов. Бесклассовая междоменная маршрутизация (Classless Interdomain Routing, CIDR) была создана для упрощения масок подсети. Если мы возьмем предыдущий пример, то можем написать subnet /24 (эквивалент CIDR) вместо длинной записи. В табл. 3.2 (можно использовать как справочник) перечислены подсети и маски сети.

Таблица 3.2. Подсети и CIDR

CIDR	МАСКА СЕТИ	КОЛИЧЕСТВО ХОСТОВ
/30	255.255.255.252	2
/29	255.255.255.248	6
/28	255.255.255.240	14
/27	255.255.255.224	30
/26	255.255.255.192	62
/25	255.255.255.128	126
/24	255.255.255.0	254
/23	255.255.254.0	510
/22	255.255.252.0	1022
/21	255.255.248.0	2046
/20	255.255.240.0	4094
/19	255.255.224.0	8190
/18	255.255.192.0	16382
/17	255.255.128.0	32766
/16	255.255.0.0	65534
/15	255.254.0.0	131070
/14	255.252.0.0	262142
/13	255.248.0.0	524286
/12	255.240.0.0	1048574
/11	255.224.0.0	2097150
/10	255.192.0.0	4194302
/9	255.128.0.0	8288606
/8	255.0.0.0	16777216

IPv6

Пока что мы все еще активно используем IPv4 для организации сетевой инфраструктуры. Было несколько попыток перейти с IPv4 на IPv6, поскольку однажды в мире закончатся адреса IPv4. Вам нужно хотя бы понимать, как IPv6 работает на практике.

Адрес IPv6 состоит из 128 бит шестнадцатеричных символов. Ниже приведен пример IPv6, и мне не хватит и триллиона слов, чтобы все объяснить:

fff0:0000:eeee:0000:0000:0000:fe77:03aa

Мы воспользуемся этим примером, чтобы увидеть, как работает формат IPv6.

1. Чтобы проследить специфику IPv6, сначала нужно удалить ведущие нули.

   До: fff0:**0000**:eeee:**0000:0000:0000**:fe77:**0**3aa

   После: fff0:0:eeee:0:0:0:fe77:3aa

2. Сократите серию нулей (в нашем случае это три секции нулей, идущие подряд) и замените их на `::`

   До: fff0:0:eeee:**0:0:0:**fe77:3aa

   После: fff0:0:eeee::fe77:3aa

Обратите внимание, что в IPv6 вы можете сжать серию нулей только один раз.

Номера портов

Номера портов и IP-адреса подобны братьям и сестрам. Без номера порта сетевой пакет никогда не сможет достичь места назначения. Номер порта подобен гражданскому адресу. Названия улицы (IP-адреса) недостаточно, чтобы добраться до определенной квартиры; вам понадобится гражданский номер (номер порта), чтобы иметь полный и достаточный адрес.

Представьте, что используете свой браузер для перехода на www.google.com. Вашему пакету потребуются IP-адрес хоста веб-сервера и номер порта, который по умолчанию равен 443 для HTTPS. На том же сервере (с тем же IP-адресом) Google может размещать другие сервисы, например FTP; тогда пакет будет использовать порт 21 для его достижения.

В табл. 3.3 перечислены наиболее распространенные номера портов по умолчанию, которые вам необходимо знать при сканировании сети. Обратите внимание, что номера портов находятся в диапазоне от 1 до 65 535.

В этой таблице перечислены самые популярные номера портов, которые, как я считаю, вам необходимо знать. Полный список можно найти в «Википедии» по адресу en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers.

Таблица 3.3. Общие номера портов

НАЗВАНИЕ ПРОТОКОЛА	ПОРТ	НАЗВАНИЕ ПРОТОКОЛА	ПОРТ
FTP	TCP 21	LDAP поверх SSL	TCP 636
SSH/SCP	TCP 22	FTP поверх SSL	TCP 989–990
Telnet	TCP 23	IMAP поыерх SSL	TCP 993
SMTP	TCP 25	POP3 поыерх SSL	TCP 995
DNS-запрос	UDP 53	MS-SQL	TCP 1433
Передача зоны DNS	TCP 53	NFS	TCP 2049
DHCP	UDP 68	Docker Daemon	TCP 2375
TFTP	UDP 69	Oracle DB	TCP 2483–2484
HTTP	TCP 80	MySQL	TCP 3306
Kerberos	UDP 88	RDP	TCP 3389
POP3	TCP 110	VNC	TCP 5500
SNMP	UDP 161 UDP 162	PCAnywhere	TCP 5631
NetBIOS	TCP/UDP 137 TCP/UDP 138 TCP/UDP 139	IRC	TCP 6665–6669
IMAP	TCP 143	IRC SSL	TCP 6679 TCP 6697
LDAP	TCP 389	BitTorrent	TCP 6881–6999
HTTPS (TLS)	TCP 443	Принтеры	TCP 9100
SMTP поверх SSL	TCP 465	WebDAV	TCP 9800
rlogin	TCP 513	Webmin	10000

СЕТЕВОЕ СКАНИРОВАНИЕ

Теперь, когда вы понимаете основы работы в сети, пора приступить к делу. В следующих разделах вы узнаете, как определять целевые хосты в сети.

Определение живых хостов

Есть несколько способов определить, работает ли хост в сети.

Пинг

Вы можете использовать Ping, чтобы быстро проверить сетевое соединение. Итак, каковы же принципы работы Ping?

Когда вы выполняете команду `ping`, ваш хост Kali отправит эхо-запрос ICMP (echo request) в пункт назначения, а после этого цель ответит пакетом ICMP (echo reply). Таким образом, можно сказать, что цель доступна. Команда `ping` полезна для системных администраторов, но мы ведь элита, верно? Позже вы поймете, почему должны использовать Nmap для поиска активных хостов. Наконец, вы должны знать, что некоторые системные администраторы закрывают эхо-запрос ICMP на уровне брандмауэра, чтобы хакеры не могли проверить доступность некоторых серверов.

ARP

Протокол разрешения адресов (Address Resolution Protocol, ARP) — это фантастическая утилита, которая преобразует IP-адреса в физические MAC-адреса в локальной сети.

Теперь мы можем воспользоваться содержимым таблицы ARP, чтобы вывести список всех хостов в одной сети с помощью команды `arp-scan` в Kali:

```
root@kali:~# arp-scan 10.0.0.1/24
Interface: eth0, type: EN10MB, MAC: 00:0c:29:40:e7:a6, IPv4: 10.0.0.20
WARNING: host part of 10.0.0.1/24 is non-zero
Starting arp-scan 1.9.7 with 256 hosts (https://github.com/royhills/
arp-scan)
10.0.0.1        b4:fb:e4:2f:04:3d       Ubiquiti Networks Inc.
10.0.0.2        fc:ec:da:d4:d5:99       Ubiquiti Networks Inc.
10.0.0.5        b4:fb:e4:1b:c4:d2       Ubiquiti Networks Inc.
10.0.0.10       70:5a:0f:f6:fc:3a       Hewlett Packard
10.0.0.50       00:11:32:94:25:4c       Synology Incorporated
10.0.0.75       fc:ec:da:d8:24:07       Ubiquiti Networks Inc.
10.0.0.102      d0:2b:20:95:3b:96       Apple, Inc.
```

Nmap

Пришло время показать вам мой любимый инструмент Nmap, который я использую для идентификации живых хостов. Для выполнения работы вам потребуются следующие параметры команды:

```
$nmap -sn [IP-адреса / Диапазон]
```

Чтобы помочь вам запомнить их, предлагаю подумать о параметрах следующим образом: `-s` — это Сэм, а `n` — Няня. Настоящее значение этих опций таково:

- `n` означает «нет»;
- `s` означает «сканирование».

Вот почему эта опция называется «Без сканирования портов» (No Port Scan). Некоторые называют это сканированием Ping, но не путайте его с инструментом ICMP Ping, о котором мы говорили ранее в текущей главе. При этом посмотрим,

почему данный вариант волшебный. Чтобы идентифицировать живые хосты, Nmap попытается выполнить ряд действий, которые описаны ниже.

1. Отправит эхо-запрос ICMP, но Nmap не сдастся, если протокол ICMP заблокирован.
2. Кроме того, отправит запрос метки времени ICMP.
3. Отправит пакет ACK на порт 80 и пакет SYN на порт 443.
4. Наконец, отправит запрос ARP.

Он невероятно мощный, правда? Важно понимать, что вы должны быть пользователем root (или членом группы `sudo`) в вашем Kali, иначе ваши возможности будут ограничены и вы не сможете выполнять все эти функции. Приступим к действию Сэма и Няни:

```
root@kali:~# nmap -sn 10.0.0.1/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-05 09:25 EDT
Nmap scan report for USGPRO (10.0.0.1)
Host is up (0.00036s latency).
MAC Address: B4:FB:E4:2F:04:3D (Ubiquiti Networks)
Nmap scan report for unifi (10.0.0.2)
Host is up (0.00027s latency).
MAC Address: FC:EC:DA:D4:D5:99 (Ubiquiti Networks)
Nmap scan report for 10.0.0.5
Host is up (0.0024s latency).
MAC Address: B4:FB:E4:1B:C4:D2 (Ubiquiti Networks)
Nmap scan report for 10.0.0.10
Host is up (0.0081s latency).
MAC Address: 70:5A:0F:F6:FC:3A (Hewlett Packard)
Nmap scan report for 10.0.0.50
Host is up (0.00066s latency).
MAC Address: 00:11:32:94:25:4C (Synology Incorporated)
```

Сканирование портов и перечисление сервисов

Одна из задач, которую вам нужно будет решать во время сканирования сети, — это поиск открытых портов на каждом хосте. Зачем? Допустим, вы хотите знать все веб-серверы в локальной сети (local area network, LAN); сканирование портов позволит вам легко получить эту информацию. Посмотрим, как Nmap играючи справится с данной задачей.

SYN-сканирование портов TCP

В Nmap так много опций для выполнения сканирования портов, но я всегда использую для TCP сканирование SYN. Фактически Nmap по умолчанию будет выполнять этот тип сканирования портов:

```
$nmap -sS [IP-адреса / Диапазон]
```

Чтобы запомнить это, вы можете соотнести параметр `-sS` с Сэмом и Самантой. Всегда думайте о Сэме и Саманте, когда хотите выполнить сканирование портов. Вам повезло, если вас зовут Сэм, но параметр `sS` означает *SYN-сканирование*, и некоторые люди называют его *скрытым сканированием*. Я придумал ассоциацию с Сэмом и Самантой, чтобы вы могли легко запомнить параметр.

Позвольте мне объяснить вам, как работает SYN-сканирование в Nmap. Сканер, запущенный с опцией `sS`, отправит SYN-запрос на сервер и, получив в ответ SYN/ACK, покажет, что порт открыт. Если же сканер не получил SYN/ACK, то порт либо закрыт, либо фильтруется. Для справки: *фильтруется* означает, что его защищает сетевой экран:

```
root@kali:~# nmap -sS 10.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-05 09:27 EDT
Nmap scan report for USGPRO (10.0.0.1)
Host is up (0.00051s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
53/tcp  open  domain
80/tcp  open  http
443/tcp open  https
MAC Address: B4:FB:E4:2F:04:3D (Ubiquiti Networks)

Nmap done: 1 IP address (1 host up) scanned in 0.25 seconds
```

UDP

А что насчет сканирования портов UDP? Чтобы использовать сканирование портов UDP в Nmap, вы должны добавить ключ `sU`:

```
$nmap -sU [IP-адреса / Диапазон]
```

Важно отметить: UDP работает медленно из-за того, что не требует установления соединения. Мы всегда можем использовать параметр интервалов времени T5, чтобы настроить сканирование и ускорить его. Брандмауэры могут легко заблокировать сканирование с T5 для интернет-адреса. Опция T2 — ваш друг при сканировании IP-адресов в интернете, поэтому вы можете обойти механизмы выявления сканирования (брандмауэры и т. д.):

```
$nmap -sU -T5 [IP-адреса / Диапазон]
```

Итак, чтобы UDP-сканер определил, открыт порт или закрыт, он отправит UDP-пакет и будет ждать ответа от пункта назначения. Если Nmap получил ответ или ответа не последовало, то, возможно, порт открыт.

В то же время если сканер получил ошибку ICMP, то порт либо закрыт, либо фильтруется:

```
root@kali:~# nmap -sU -T5 10.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-05 09:28 EDT
Warning: 10.0.0.1 giving up on port because retransmission cap hit (2).
Nmap scan report for USGPRO (10.0.0.1)
Host is up (0.0014s latency).
Not shown: 875 open|filtered ports, 123 closed ports
PORT    STATE SERVICE
53/udp  open  domain
123/udp open  ntp
MAC Address: B4:FB:E4:2F:04:3D (Ubiquiti Networks)

Nmap done: 1 IP address (1 host up) scanned in 119.25 seconds
```

Основы использования сканера Nmap

Обсудим несколько основ Nmap. Если вы запустите Nmap без каких-либо опций, то инструмент по умолчанию будет использовать три важные функции:

- установит скорость T3;
- просканирует 1000 самых распространенных портов TCP;
- предполагая, что вы являетесь пользователем root в вашей Kali, по умолчанию установит сканирование SYN TCP.

Другими словами, все происходит в фоновом режиме. Это значит, что вам не нужно указывать скорость T3, поскольку она установлена по умолчанию. Это так же работает и для номеров портов (вам не нужно добавлять `--top-ports 1000`) или TCP SYN Scan (вам не нужно добавлять параметр `-sS`). В предыдущих примерах мы указали параметр `sS` для сканирования SYN, но здесь нам не нужно этого делать, поскольку Nmap установит его по умолчанию, верно?

Что касается настройки, то всегда продуманно выбирайте скорость. Например, не используйте высокую скорость, такую как T5, на рабочем IP-адресе; вместо этого придерживайтесь значения по умолчанию (T3). Кроме того, убедитесь, что вы выбрали количество портов, которое соответствует вашим потребностям: либо 100 самых распространенных, либо вариант по умолчанию — 1000. Посмотрим на практический пример; допустим, вы хотите сканировать только 100 портов с помощью сканирования TCP:

```
#Более быстрое сканирование TCP
$nmap –-top-ports 100 -T5 [IP-адреса / Диапазон]
```

Если вы нацеливаетесь на конкретный порт, то используйте параметр `-p`, за которым следует номер порта, диапазон или список:

```
#Сканирование для поиска HTTP (порт 80) в сети
$nmap -p 80 [IP-адреса / Диапазон]
```

Наконец, если вы хотите включить все порты, то используйте `-p-` (также вы можете использовать `-p 1-65535`) для сканирования каждого порта (сканирование всех номеров портов выявит все скрытые приложения). Я никогда не использую этот параметр для UDP (он слишком медленный), но часто применяю его для сканирования TCP (в следующей команде мы не указали параметр `-sS`, поскольку он есть по умолчанию):

```
root@kali:~# nmap -p- -T5 10.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-05 09:35 EDT
Nmap scan report for USGPRO (10.0.0.1)
Host is up (0.00097s latency).
Not shown: 65531 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
53/tcp  open  domain
80/tcp  open  http
443/tcp open  https
MAC Address: B4:FB:E4:2F:04:3D (Ubiquiti Networks)

Nmap done: 1 IP address (1 host up) scanned in 9.91 seconds
```

Перечисление сервисов

Пришло время посмотреть, как выполнить сканирование версии сервиса с помощью лучшего сканера: Nmap. Ключом для сканирования версий сервисов в Nmap является `-sV`. Как правило, хороший способ запомнить параметры команды в Nmap — придумывать значение каждой букве. Например, `s` означает *сканирование*, а `V` означает *версию*. Легко, правда? Вот почему это называется *сканированием версии*. Обратите внимание, что сканирование версии займет больше времени, чем обычное сканирование портов, поскольку сканер будет пытаться определить тип сервиса. В следующем примере мы просканируем тот же хост, который использовали ранее, однако на этот раз добавим параметр `-sV` (проверьте столбец версии):

```
root@kali:~# nmap -p- -T5 -sV 10.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-05 09:36 EDT
Nmap scan report for USGPRO (10.0.0.1)
Host is up (0.00097s latency).
Not shown: 65531 closed ports
PORT    STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 6.6.1p1 Debian 4~bpo70+1 (protocol 2.0)
53/tcp  open  domain   dnsmasq 2.78-23-g9e09429
80/tcp  open  http     lighttpd
443/tcp open  ssl/http Ubiquiti Edge router httpd
MAC Address: B4:FB:E4:2F:04:3D (Ubiquiti Networks)
Service Info: OS: Linux; Device: router; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.88 seconds
```