

Scanning and Identifying Network Nodes

Scenario

In this activity, you will use common network tools, such as local commands and the Nmap network mapper, to discover other hosts on the local network. Next, you will conduct a banner grabbing exercise to identify specific services on the hosts. Finally, you will use DNS tools to gather name resolution information.

Objectives

This activity is designed to test your understanding of and ability to apply content examples in the following CompTIA Security+ objective:

- 4.1 Given a scenario, use the appropriate tool to assess organizational security.

Lab

- Kali VM
- DC1 VM
- pfSense VM

Task 1

Identify local network configuration

Determine the configuration of the local host and its subnet, using tools such as **ifconfig**, **ip**, **arp**, **netdiscover**, and **pathping**. Run the scans from **Kali** VM and the **DC1** VM.

1. On the **Kali** VM, sign in as **kali** using **Pa\$\$w0rd** as the password.
2. From the top bar, select the **Terminal Emulator** icon.
3. Run the **ifconfig** command to display the interface configuration. Note the IP address assigned to the **eth0** interface.

```
ifconfig eth0
```

```

(kali㉿kali)-[~]
$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.12 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::e39:47ff:fe55:0 prefixlen 64 scopeid 0x20<link>
    ether 0c:39:47:55:00:00 txqueuelen 1000 (Ethernet)
    RX packets 10711 bytes 1858059 (1.7 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10533 bytes 1413688 (1.3 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Figure 1.1 – The `ifconfig eth0` command.

4. Run the **ip a** command to display the same information using the newer **ip a** tool.

`ip a`

```

(kali㉿kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:39:47:55:00:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.12/24 brd 192.168.1.255 scope global dynamic eth0
        valid_lft 6536sec preferred_lft 6536sec
    inet6 fe80::e39:47ff:fe55:0/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

```

Figure 1.2 – The `ip a` command.

5. Run the following command to identify the default gateway.

`ip route show`

```

(kali㉿kali)-[~]
$ ip route show
default via 192.168.1.254 dev eth0 proto dhcp src 192.168.1.10 metric 100
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.10 metric 100

```

Figure 1.3 – The `ip route show` command and default gateway.

Because the network uses DHCP to provide client addresses, the local machine has been configured with a default gateway addresses automatically.

6. Run the **arp -a** command to check the ARP cache to display other hosts local to the subnet.

`arp -a`

```

(kali㉿kali)-[~]
$ arp -a
pfsense.contoso.local (192.168.1.254) at 0c:9e:d7:84:00:01 [ether] on eth0
dc1.contoso.local (192.168.1.1) at 0c:94:de:ef:00:00 [ether] on eth0

```

Figure 1.4 – The arp -a command.

7. Run the **ip neighbor** command to display similar information using the newer **ip tool**.

`ip neighbor`

```
(kali㉿kali)-[~]  
$ ip neighbor  
192.168.1.254 dev eth0 lladdr 0c:9e:d7:84:00:01 STALE  
192.168.1.1 dev eth0 lladdr 0c:94:de:ef:00:00 REACHABLE
```

Figure 1.5 – The ip neighbor command.

NOTE: The ARP cache shows only machines that have communicated with the local host. To verify whether any other hosts are present, you can perform a “sweep” of the local network. One means of doing this is to use **ping** in a for/next loop. You can also use the **netdiscover** tool bundled with Kali.

8. Run the following command with **sudo** to scan the network by using **netdiscover**. The results should discover several other hosts connected to the local virtual switch.

`sudo netdiscover -i eth0 -r 192.168.1.0/24`

```
Currently scanning: Finished! Screen View: Unique Hosts  
3 Captured ARP Req/Rep packets, from 2 hosts. Total size: 180  
+-----+-----+-----+-----+-----+  
IP           At MAC Address      Count  Len  MAC Vendor / Hostname  
+-----+-----+-----+-----+-----+  
192.168.1.1   0c:94:de:ef:00:00    2     120  Unknown vendor  
192.168.1.254 0c:9e:d7:84:00:01    1      60  Unknown vendor  
+-----+-----+-----+-----+-----+
```

Figure 1.6 – The netdiscover command.

TIP: The netdiscover may take up to two minutes to complete after it reports as **Finished**.

9. Press **q** to **exit** the netdiscover report and return to the command prompt.

TIP: Run `netdiscover -h` to view the help page. The tool can operate in a passive mode, but you do not need to be stealthy, so you will run an active scan.

10. Switch to the **DC1** VM and then login as **CONTOSO\Administrator** using **Pa\$\$w0rd** as the password.

11. Right-click the **Start** menu and select **Windows PowerShell (Admin)**.

NOTE: If prompted, confirm UAC by selecting **Yes**.

12. Run the following command to display the IP address configuration for DC1.

`ipconfig`

```

PS C:\Users\Administrator> ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::2035:fe95:3555:4e6c%5
    IPv4 Address. . . . . : 192.168.1.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.254
PS C:\Users\Administrator>

```

Figure 1.7 – The ipconfig command.

13. Run the following command to test the reliability (packet loss) and latency (delay) of the connection between the **DC1** VM and the **Kali** VM (the test takes 30-45 seconds to run).

pathping 192.168.1.12

```

PS C:\Users\Administrator> pathping 192.168.1.12

Tracing route to 192.168.1.12 over a maximum of 30 hops

  0  DC1.contoso.local [192.168.1.1]
  1  192.168.1.12

Computing statistics for 25 seconds...
Hop  RTT      Source to Here   This Node/Link   Address
     0                               DC1.contoso.local [192.168.1.1]
     1    1ms      0/ 100 = 0%      0/ 100 = 0%      192.168.1.12

Trace complete.
PS C:\Users\Administrator>

```

Figure 1.8 – The pathping command.

NOTE: The two machines are not very far apart on the network. They are on the same subnet. If you run **pathping** against network nodes that have several routers between them, the utility might display lost packets along the route, which helps to clarify where network communications might be unreliable. For example, from a Windows computer with Internet access, you could run **pathping 8.8.8.8** to trace the route to one of Google's routers.

Task 2

Use nmap to discover hosts

From a penetration tester or threat actor perspective, network reconnaissance will typically aim to discover the following:

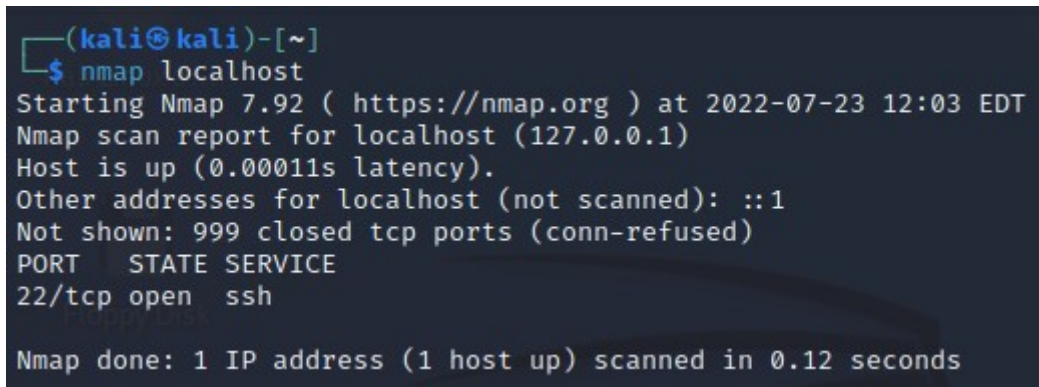
- Default gateway (the router connecting the subnet to other networks).
- DNS server (used to resolve host names on the network).
- Whether any network directory/authentication and application servers are present.
- Whether any host/client access devices are present.

- Whether any other types of devices (embedded systems or appliances) are present.

Use Nmap from the **Kali** VM to report information for this network.

1. Switch to **Kali** VM. If necessary, log in as user **kali** using **Pa\$\$w0rd** as the password.
2. Run the following command to scan the **Kali** VM:

`nmap localhost`

A terminal window with a dark background and light blue text. The prompt is (kali@kali)-[~]. The command \$ nmap localhost has been entered. The output shows the Nmap version (7.92), the target (localhost/127.0.0.1), and the scan results: host is up, one open port (22/tcp) for ssh, and 999 closed ports. The scan took 0.12 seconds.

```
(kali@kali)-[~]  
$ nmap localhost  
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:03 EDT  
Nmap scan report for localhost (127.0.0.1)  
Host is up (0.00011s latency).  
Other addresses for localhost (not scanned): ::1  
Not shown: 999 closed tcp ports (conn-refused)  
PORT      STATE SERVICE  
22/tcp    open  ssh  
  
Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
```

Figure 2.1 – The nmap command.

3. Run the following command to do a basic network scan.

`nmap 192.168.1.0/24`


```

(kali@kali)-[~]
$ nmap 192.168.1.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:12 EDT
Nmap scan report for DC1.contoso.local (192.168.1.1)
Host is up (0.0042s latency).
Not shown: 987 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
5357/tcp  open  wsapi

Nmap scan report for 192.168.1.12
Host is up (0.00013s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap scan report for pfsense.contoso.local (192.168.1.254)
Host is up (0.0044s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http

Nmap done: 256 IP addresses (3 hosts up) scanned in 8.90 seconds

```

Figure 2.2 – Using nmap to map a network.

4. Run the following command and check the output. What services are running and what do they tell you about the host?

```
sudo nmap -sS 192.168.1.254
```

```

(kali@kali)-[~]
$ sudo nmap -sS 192.168.1.254
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:16 EDT
Nmap scan report for pfsense.contoso.local (192.168.1.254)
Host is up (0.0019s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
MAC Address: 0C:9E:D7:84:00:01 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 4.96 seconds

```

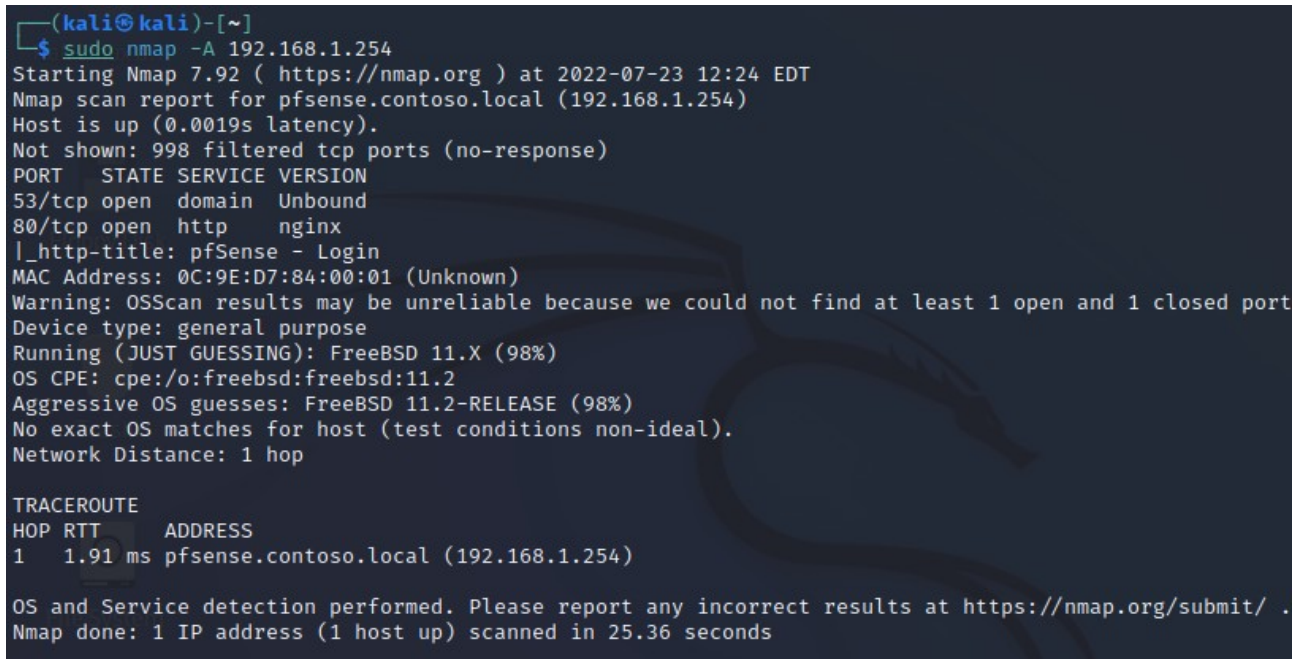
Figure 2.3 – Nmap SYN scan.

This syntax will scan the default port range (most commonly used 1000 ports) on the target.

TIP: The -sS option in nmap performs a TCP SYN scan or half open connection scan.

5. Run the following command to identify more about the host:

```
sudo nmap -A 192.168.1.254
```

A terminal window with a dark background and light-colored text. The prompt is (kali@kali)-[~]. The command sudo nmap -A 192.168.1.254 has been executed. The output shows the Nmap version (7.92), the target host (pfsense.contoso.local), and various scan results including open ports (53/tcp, 80/tcp), service versions (Unbound, nginx), and OS detection (FreeBSD 11.X). It also includes a traceroute and a final summary of the scan.

```
(kali@kali)-[~]
$ sudo nmap -A 192.168.1.254
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:24 EDT
Nmap scan report for pfsense.contoso.local (192.168.1.254)
Host is up (0.0019s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
53/tcp    open  domain Unbound
80/tcp    open  http   nginx
|_http-title: pfSense - Login
MAC Address: 0C:9E:D7:84:00:01 (Unknown)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): FreeBSD 11.X (98%)
OS CPE: cpe:/o:freebsd:freebsd:11.2
Aggressive OS guesses: FreeBSD 11.2-RELEASE (98%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1   1.91 ms  pfsense.contoso.local (192.168.1.254)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.36 seconds
```

Figure 2.4 – The nmap -A command.

NOTE: The -A option enables OS detection, version detection, script scanning, and traceroute.

6. Run the following command to scan for open ports between 20-200 on the network:

```
sudo nmap -p 20-200 192.168.1.0/24
```

```

(kali㉿kali)-[~]
└─$ sudo nmap -p 20-200 192.168.1.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:30 EDT
Nmap scan report for DC1.contoso.local (192.168.1.1)
Host is up (0.0030s latency).
Not shown: 176 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
MAC Address: 0C:94:DE:EF:00:00 (Unknown)

Nmap scan report for pfsense.contoso.local (192.168.1.254)
Host is up (0.0019s latency).
Not shown: 179 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
MAC Address: 0C:9E:D7:84:00:01 (Unknown)

Nmap scan report for 192.168.1.10
Host is up (0.0000070s latency).
Not shown: 180 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap scan report for 192.168.1.12
Host is up (0.0000090s latency).
Not shown: 180 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap done: 256 IP addresses (4 hosts up) scanned in 4.85 seconds

```

Figure 2.5 – Nmap scanning for open ports between 20-200.

7. Run the following command to scan for the **ten** most common ports:

```
sudo nmap --top-ports 10 192.168.1.0/24
```



```

(kali㉿kali)-[~]
└─$ sudo nmap --top-ports 10 192.168.1.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:33 EDT
Nmap scan report for DC1.contoso.local (192.168.1.1)
Host is up (0.0030s latency).

PORT      STATE      SERVICE
21/tcp    filtered  ftp
22/tcp    filtered  ssh
23/tcp    filtered  telnet
25/tcp    filtered  smtp
80/tcp    open      http
110/tcp   filtered  pop3
139/tcp   open      netbios-ssn
443/tcp   filtered  https
445/tcp   open      microsoft-ds
3389/tcp  filtered  ms-wbt-server
MAC Address: 0C:94:DE:EF:00:00 (Unknown)

Nmap scan report for pfsense.contoso.local (192.168.1.254)
Host is up (0.0018s latency).

PORT      STATE      SERVICE
21/tcp    filtered  ftp
22/tcp    filtered  ssh
23/tcp    filtered  telnet
25/tcp    filtered  smtp
80/tcp    open      http
110/tcp   filtered  pop3
139/tcp   filtered  netbios-ssn
443/tcp   filtered  https
445/tcp   filtered  microsoft-ds
3389/tcp  filtered  ms-wbt-server
MAC Address: 0C:9E:D7:84:00:01 (Unknown)

```

Figure 2.6 – Nmap scan of the top 10 ports on the network.

8. Run the following command to quickly scan the network for hosts that are **up** or **down** on the network.

```
nmap -sn 192.168.1.0/24
```

```

(kali㉿kali)-[~]
└─$ nmap -sn 192.168.1.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-23 12:38 EDT
Nmap scan report for DC1.contoso.local (192.168.1.1)
Host is up (0.0047s latency).
Nmap scan report for 192.168.1.10
Host is up (0.0013s latency).
Nmap scan report for 192.168.1.12
Host is up (0.013s latency).
Nmap scan report for pfsense.contoso.local (192.168.1.254)
Host is up (0.0025s latency).
Nmap done: 256 IP addresses (4 hosts up) scanned in 2.69 seconds

```

Figure 2.7 – Nmap scanning for live hosts.

Task 3

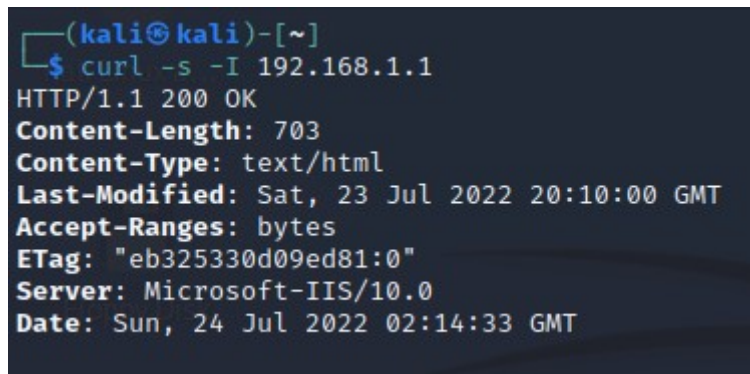
Banner grab with curl and Firefox

Banner grabbing is a way of identifying service versions. This information helps attackers select potentially vulnerable machines. The information also helps administrators confirm that all services on the network are of a certain version.

Basic tools can be used to grab banners. In this activity, you will use **cURL** and **Firefox**.

1. Run the following command in the **Terminal** to connect to the 192.168.1.1 HTTP server by using cURL.

```
curl -s -I 192.168.1.1
```



```
(kali㉿kali)-[~]  
$ curl -s -I 192.168.1.1  
HTTP/1.1 200 OK  
Content-Length: 703  
Content-Type: text/html  
Last-Modified: Sat, 23 Jul 2022 20:10:00 GMT  
Accept-Ranges: bytes  
ETag: "eb325330d09ed81:0"  
Server: Microsoft-IIS/10.0  
Date: Sun, 24 Jul 2022 02:14:33 GMT
```

Figure 3.1 – The cURL command.

NOTE: You used port scans in the earlier activities to determine that the 192.168.1.1 virtual machine is running a web service. With this cURL command, you have determined the software type and version for the web server service.

2. In the **Terminal** type the following and press **Enter**.

```
firefox http://192.168.1.1 &
```

3. Close the **Firefox** web browser.

Task 4

Query DNS

DNS provides name resolution to internal networks as well as the Internet. DNS is used any time a user or application refers to a host by name. DNS queries name records to find the IP address associated with a hostname or fully qualified domain name (FQDN). These name records can also reveal information about how a network is configured. In this task, you will gather DNS information by using the **dig** utility.

1. Run the **dig** command in the **Terminal** to perform a reverse lookup on the default gateway.

dig -x 192.168.1.254

```
(kali㉿kali)-[~]
$ dig -x 192.168.1.254

; <<>> DiG 9.18.1-1-Debian <<>> -x 192.168.1.254
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 6891
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;254.1.168.192.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
254.1.168.192.in-addr.arpa. 3600 IN      PTR      pfsense.contoso.local.

;; Query time: 0 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Jul 23 18:20:26 EDT 2022
;; MSG SIZE rcvd: 90
```

Figure 4.1 – The dig command.

2. Run the **dig** command in the **Terminal** to perform a name resolution on server **dc1.contoso.local**

dig dc1.contoso.local

```
(kali㉿kali)-[~]
$ dig dc1.contoso.local

; <<>> DiG 9.18.1-1-Debian <<>> dc1.contoso.local
;; global options: +cmd
;; Got answer:
;; WARNING: .local is reserved for Multicast DNS
;; You are currently testing what happens when an mDNS query is leaked to DNS
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 53570
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;dc1.contoso.local.              IN      A

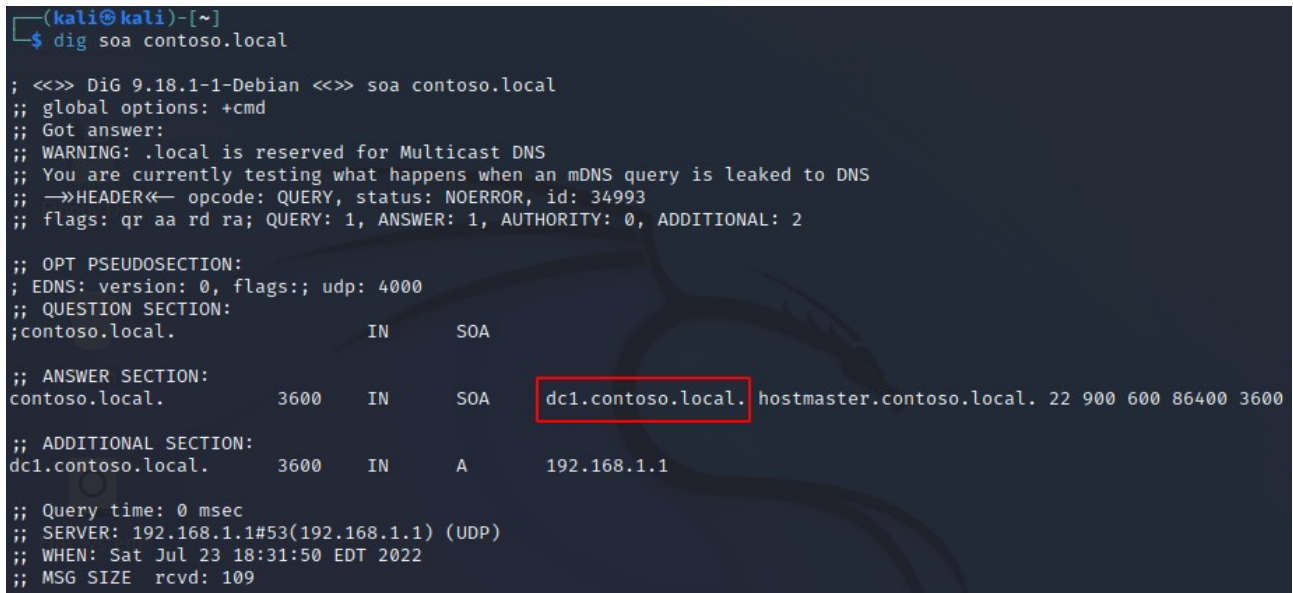
;; ANSWER SECTION:
dc1.contoso.local.      3600    IN      A      192.168.1.1

;; Query time: 4 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Jul 23 18:27:48 EDT 2022
;; MSG SIZE rcvd: 62
```

Figure 4.2 – The dig command name resolution.

3. Run the **dig** command in the **Terminal** to display the authoritative DNS server for the namespace.

```
dig soa contoso.local
```



```
(kali@kali)-[~]
$ dig soa contoso.local

;<<>> DiG 9.18.1-1-Debian <<>> soa contoso.local
;; global options: +cmd
;; Got answer:
;; WARNING: .local is reserved for Multicast DNS
;; You are currently testing what happens when an mDNS query is leaked to DNS
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 34993
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;contoso.local.                IN      SOA

;; ANSWER SECTION:
contoso.local.                3600    IN      SOA      dc1.contoso.local. hostmaster.contoso.local. 22 900 600 86400 3600

;; ADDITIONAL SECTION:
dc1.contoso.local.            3600    IN      A        192.168.1.1

;; Query time: 0 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Sat Jul 23 18:31:50 EDT 2022
;; MSG SIZE rcvd: 109
```

Figure 4.3 – The SOA records.

NOTE: The query returns the FQDN of the DNS server responsible for the domain (dc1.contoso.local) and its host record (192.168.1.1). It's also worth noting some of the flags shown:

- **aa** indicates that the answer is authoritative. The “AUTHORITY” section of the response is empty. Contents for this section are commonly omitted by name servers to reduce the size of responses.
- **ra** indicates that recursion is available, that is, this router will forward queries to other servers.

4. Close the **Terminal** window by typing the following:

```
exit
```