Mia Weber
CSCI 465 - Net/App Security
09-04-23

A1- Try Hack Me Exercises

Table of Contents

Passive Recon

Introduction:

- We use *whois* to query WHOIS records, *nslookup* and *dig* to query DNS database records. All of these are publicly available records and don't alert the target.

Passive vs Active Recon:

- Attacks need to gather information about target systems.
- Defenders need to know what an adversary will discover about your systems and networks.
- **Reconnaissance (recon) –** a preliminary survey to gather information about a target. First step in The Unified Kill Chain to gain initial foothold on system. Two types of recon: passive recon and active recon.
- **Passive Reconnaissance –** rely on publicly available knowledge that you can access from publicly available resources without directly engaging with the target.
    - Ex: Looking up DNS records of a domain from a public DNS server, checking job ads related to the target, reading news articles about the target company.
- **Active Reconnaissance –** isn't achieved so discreetly, requires direct engagement with the target (like checking the locks on the doors and windows and other entry points). Because it is invasive, you need proper legal authorization to avoid legal trouble.
    - Ex: connecting to one of the company servers (HTTP, FTP, SMTP), social engineering, entering company premises pretending to be a repairman.

Whois:

- **WHOIS –** a request and response protocol that follows the FRC 3912 specification. A WHOIS server listens on TCP port 43 for incoming requests. The *domain registrar* maintains the WHOIS records for the domain names it leases. WHOIS server replies with information related to the domain requested.
    - We can learn which registrar the domain was registered with, contact info of the registrant (name, organization, address, phone, etc.), creation, update, and expiration dates, and the name server (which server to ask to resolve the domain name).
- Use a local whois client to get this information.
    - *Whois DOMAIN_NAME* (Ex: whois tryhackme.com).

```
(base) ┌──(kali㉿kali-linux-2022-2)-[~]
└─$ whois tryhackme.com
   Domain Name: TRYHACKME.COM
   Registry Domain ID: 2282723194_DOMAIN_COM-VRSN
   Registrar WHOIS Server: whois.namecheap.com
   Registrar URL: http://www.namecheap.com
   Updated Date: 2021-05-01T19:43:23Z
   Creation Date: 2018-07-05T19:46:15Z
   Registry Expiry Date: 2027-07-05T19:46:15Z
   Registrar: NameCheap, Inc.
   Registrar IANA ID: 1068
   Registrar Abuse Contact Email: abuse@namecheap.com
   Registrar Abuse Contact Phone: +1.6613102107
   Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
   Name Server: KIP.NS.CLOUDFLARE.COM
   Name Server: UMA.NS.CLOUDFLARE.COM
   DNSSEC: unsigned
   URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2023-08-27T16:58:13Z <<<

For more information on Whois status codes, please visit https://icann.org/epp
```

- First, redirected to namecheap.com to get information (namecheap.com is maintaining the WHOIS record for the domain name).
- Depending on the scope of the pentest, you might consider an attack against the email server for the admin user of the DNS servers.
- Many WHOIS services take measures against using WHOIS queries to harvest email addresses.

Nslookup and dig:

- A – Ipv4 Addresses
- AAAA – Ipv6 Addresses
- CNAME – Canonical Name
- MX – Mail Servers
- SOA – State of Authority
- TXT – Text Records
- Ex: *nslookup -type=A tryhackme.com 1.1.1.1* can be used to return all the IPv4 addresses used by tryhackme.com
- For more advanced DNS queries and additional functionality, you can use dig (Domain Information Groper)

```
(base) ┌──(kali㉿kali-linux-2022-2)-[~]
└─$ nslookup -type=txt thmlabs.com
Server:         10.211.55.1
Address:        10.211.55.1#53

Non-authoritative answer:
thmlabs.com     text = "THM{a5b83929888ed36acb0272971e438d78}"

Authoritative answers can be found from:
```

DNSDumpster:

- DNS lookup tools (nslookup and dig) can't find subdomains on their own (subdomains can often be poorly updated and therefore are vulnerable).
- We can use an online service that offers detailed answers to DNS queries such as DNSDumpster (it will return any and all subdomains and the collected DNS information in easy-to-read tables and a graph and any collected information about listening servers).

Host Records (A) ** this data may not be current as it uses a static database (updated monthly)

tryhackme.com HTTP: cloudflare	104.22.54.228	CLOUDFLARENET unknown
www.tryhackme.com HTTP: cloudflare	172.67.27.10	CLOUDFLARENET United States
blog.tryhackme.com HTTP: cloudflare	172.67.27.10	CLOUDFLARENET United States
remote.tryhackme.com HTTP: cloudflare	172.67.27.10	CLOUDFLARENET United States
admin.tryhackme.com HTTP: cloudflare	104.22.55.228	CLOUDFLARENET unknown
help.tryhackme.com HTTP: cloudflare	172.67.27.10	CLOUDFLARENET United States

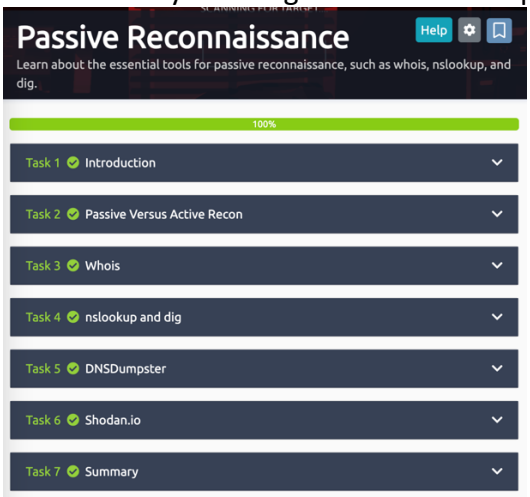
Shodan.io:

- When tasked to run a pentest against specific targets in passive recon, a service like *shodan.io* can be helpful to learn various pieces of information about the client's network without connecting to it (which would be active recon).
    - From defensive perspective, you can learn about connected and exposed devices belonging to your organization.
- *Shodan.io* tries to connect to every device reachable online to build a search engine of connected "things" in contrast with a search engine for web pages.
- We can learn things like: IP address, hosting company, geographic location, and server type and version.

Purpose	Commandline Example
Lookup WHOIS record	`whois tryhackme.com`
Lookup DNS A records	`nslookup -type=A tryhackme.com`
Lookup DNS MX records at DNS server	`nslookup -type=MX tryhackme.com 1.1.1.1`
Lookup DNS TXT records	`nslookup -type=TXT tryhackme.com`
Lookup DNS A records	`dig tryhackme.com A`
Lookup DNS MX records at DNS server	`dig @1.1.1.1 tryhackme.com MX`
Lookup DNS TXT records	`dig tryhackme.com TXT`

Summary:

By completing this lesson, I learned more information about the definitions of passive and active recon as well as got some good examples of each, learned how to do several Nmap scans including the Nmap Live Host Discovery Scan, the Nmap Basic Port Scan, Nmap Advanced Port Scan, and Nmap Post Port Scans. I also got a good review of protocols and servers as well as some security challenges that often are prevalent in a networking environment.



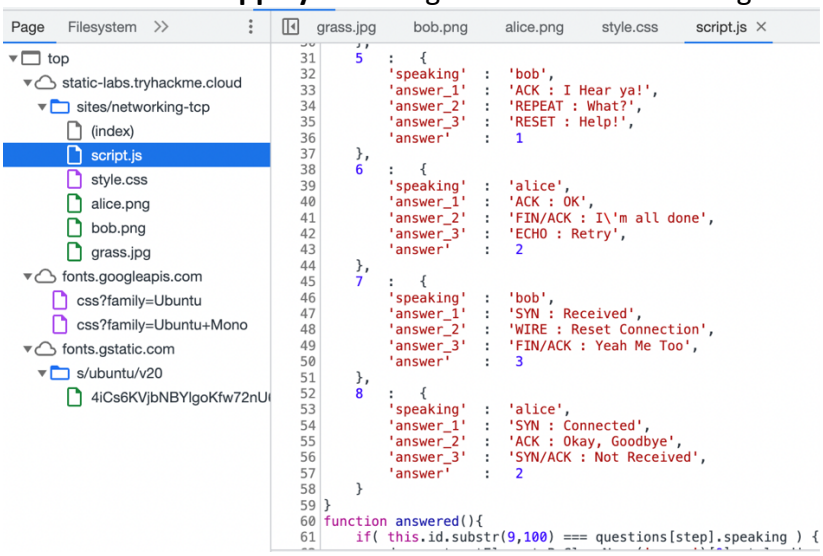
Active Recon

Introduction:

- Unlike passive recon, *Active Reconnaissance* requires you to make some kind of contact with your target (phone call, visit to company, social engineering, direct connection to target system, checking for open ports, etc.).
- It is possible to let your active reconnaissance appear as regular client activity (no one would suspect a browser connected to a target web server among hundreds of other legitimate users).

Web Browser:

- On the transport level, the browser connects to TCP port 80 by default when accessed over HTTP and TCP port 443 by default when accessed over HTTPS (it is possible to connect with custom ports which will then be listed in the address bar).
- Developer tools allow you to get information about things that your browser has received and exchanged with the remote server (ex: view and modify JS files, inspect the cookies on your system, and discover the folder structure of the site content).
- Also, helpful add-ons that provide more pen testing information:
    - **FoxyProxy –** quickly change the proxy server being used to access the target website.
    - **User-Agent Switcher and Manager –** ability to pretend to be accessing the webpage from a different operating system or web browser.
    - **Wappalyzer –** insights about the technologies used on the visited websites.



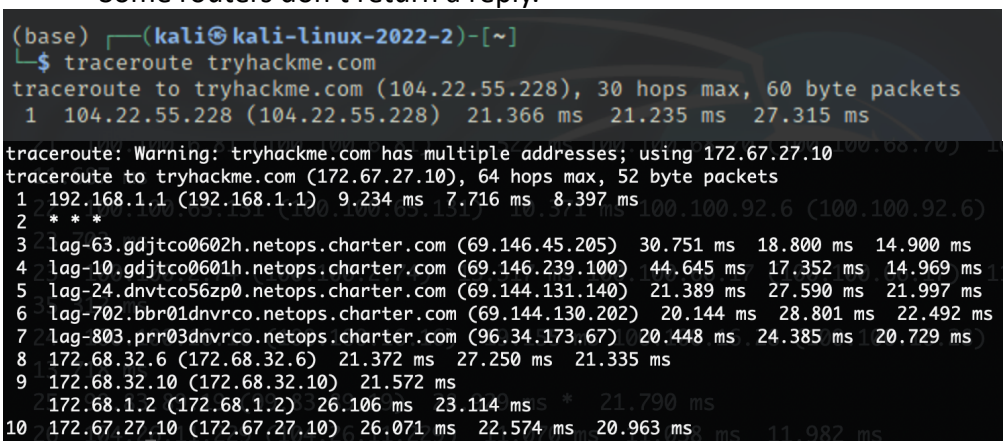
Ping:

- The primary purpose of ping is the check whether you can reach the remote system and that the remote system can reach you back – used to check network connectivity and whether the remote system is online.
- The ping command sends a packet to a remote system, and the remote system replies (sends an ICMP Echo packet to a remote system, If the system is online and the ping packet was not blocked by a firewall, the remote system will send back and ICMP Echo Reply).
- Ping falls under the protocol ICMP (Internet Control Message Protocol). ICMP supports many types of queries.

- When we don't get a ping reply back it could be because:
    - Destination computer is not responsive – still booting, turned off, OS crashed.
    - Unplugged from the network, a faulty network device across the path.
    - Firewall is configured to block some packets. Firewall might be a piece of software running on the system itself or a separate network appliance. (MS Windows firewall blocks ping by default).
    - Your system is unplugged from the network.

Traceroute:

- **Traceroute –** traces the route taken by the packets from your system to another host. The purpose is to find the IP addresses of the routers or hops that a packet traverses as it goes from your system to a target host. Also reveals the number of routers between the two systems (the route taken by the packets might change as many routers use dynamic routing protocols that adapt to network changes).
- *Traceroute 10.10.54.82*
- No direct way to discover the path from your system to a target system, rely on ICMP to "trick" the routers into revealing their IP addresses. Use a small Time To Live (TTL) in the IP header field.
    - The TTL is decreased with every router that is used, if the TTL reaches 0 the packet will be dropped and an ICMP Time-to-Live exceeded would be sent to the original sender (some routers are configured not to send such ICMP messages when discarding a packet).
- Traceroute starts by sending UDP datagrams within IP packets of TTL of 1 which will reveal the IP address of the first router to you, then it will send another packet with TTL of 2 which will be dropped at the second router revealing the IP address of the second router, and so on.
- Some routers return a public IP address. You might examine a few of these routers based on the scope of the intended penetration testing.
- Some routers don't return a reply.



Telnet:

- The TELNET (Teletype Network) protocol was developed to communicate with a remote system via a CLI (port 23). telnet sends all data, including username and passwords in cleartext which is a security concern. The secure alternative is SSH.

- Knowing that telnet client relies on the TCP protocol, you can use Telnet to connect to any service and grab its banner. Ex: *telnet 10.10.54.82 80*
- If we connect to a mail server, we need to use proper commands based on the protocol, such as SMTP and POP3.

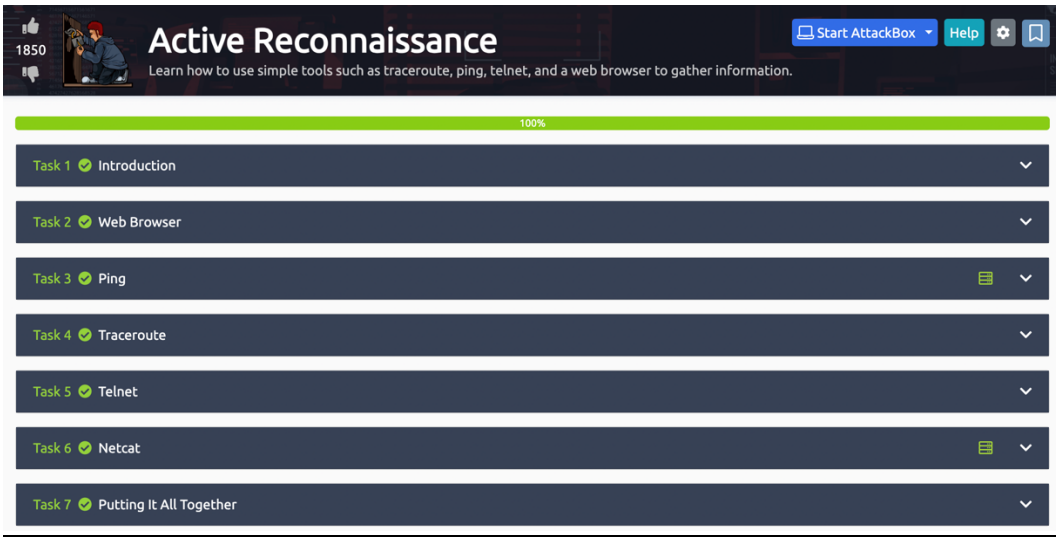
Netcat:

- Netcat (or nc) has different applications that can be of great value to a pentester. Netcat supports both TCP and UDP, it can function as a client that connects to a listening port, or it can act as a server that listens on a port of your choice.
- You can use netcat to listen on a TCP port and connect to a listening port on another system.
- Many options are supported:
    - -l listen mode
    - -p specify the port number (should appear just before the port number you want to listen on)
    - -n numeric only; no resolution of hostnames via DNS (avoids DNS lookups and warnings)
    - -v verbose output (optional, useful to discover bugs)
    - -vv very verbose (optional)
    - -k keep listening after client disconnects
- Port numbers less than 1024 require root privileges to listen on.

Summary:

By completing this lesson, I was able to focus more on active reconnaissance and the tools that are used to conduct active recon. I learned more about how to use ping, traceroute, telnet, and nc in order to gather information about the target network, system, and services that are running on the target environment.

Command	Example
ping	`ping -c 10 10.10.54.82` on Linux or macOS
ping	`ping -n 10 10.10.54.82` on MS Windows
traceroute	`traceroute 10.10.54.82` on Linux or macOS
tracert	`tracert 10.10.54.82` on MS Windows
telnet	`telnet 10.10.54.82 PORT_NUMBER`
netcat as client	`nc 10.10.54.82 PORT_NUMBER`
netcat as server	`nc -lvnp PORT_NUMBER`



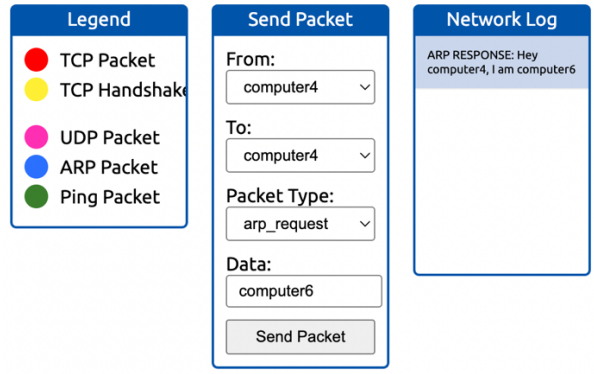
Nmap01

Introduction:

- When we target a network, we want to answer the following questions:
  - Which systems are up?
  - What services are running on those systems?
- ARP Scan: this scan uses ARP requests to discover live hosts.
- ICMP Scan: this scan uses ICMP requests to identify live hosts.
- TCP/UDP Scan: This scan sends packets to TCP ports and UDP ports to determine live hosts.
- **Nmap –** short for Network Mapper, is a free, open-source software released under GPL license and is an industry-standard tool for mapping networks, identifying live hosts, and discovering running services.

Subnetworks:

- **Network Segment –** a group of computers connected using a shared medium (ex: the Ethernet switch of WiFi access point) – physical connection.
- **Subnetwork –** the equivalent of one or more network segments connected together and configured to use the same router – logical connection.
- A subnet has its own IP address range and is connected to a more extensive network via a router (there might be a firewall enforcing security policies depending on each network).
- Subnets with /16 = 255.255.0.0 has 65 thousand hosts.
- Subnets with /24 = 255.255.255.0 has 250 hosts
- If you are connected to the same subnet, you would expect your scanner to rely on ARP (Address Resolution Protocol) queries to discover live hosts.



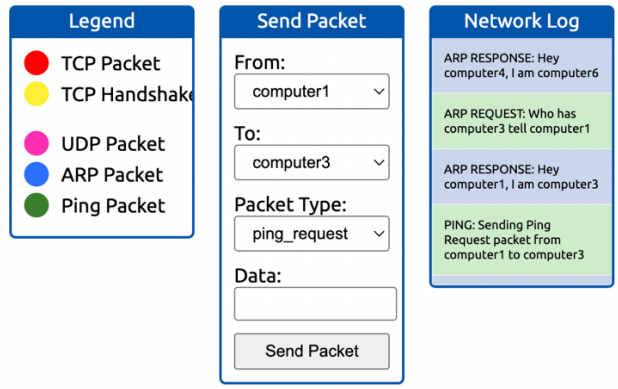
Enumerating Targets:

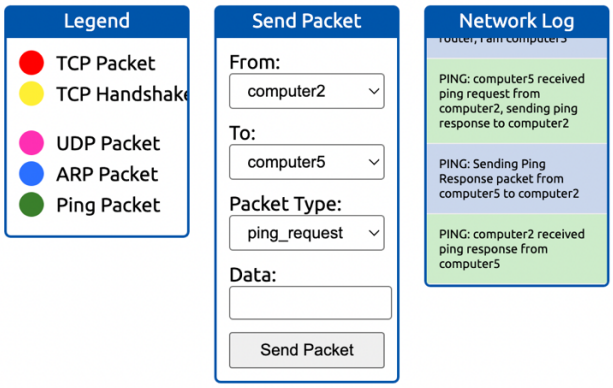
- **List –** will scan 3 IP addresses (*MACHINE_I scanme.nmap.org example.com*).
- **Range –** will scan 6 IP addresses (*10.11.12.15-20*).
- **Subnet –** will scan 4 IP addresses (*MACHINE_IP/30*).

```
(base) ┌──(kali㉿kali-linux-2022-2)-[~]
└─$ nmap -sL -n 10.10.12.13/29
Starting Nmap 7.93 ( https://nmap.org ) at 2023-08-28 18:05 MDT
Nmap scan report for 10.10.12.8
Nmap scan report for 10.10.12.9
Nmap scan report for 10.10.12.10
Nmap scan report for 10.10.12.11
Nmap scan report for 10.10.12.12
Nmap scan report for 10.10.12.13
Nmap scan report for 10.10.12.14
Nmap scan report for 10.10.12.15
Nmap done: 8 IP addresses (0 hosts up) scanned in 0.00 seconds
```

Discovering Live Hosts:

- ARP from Link Layer (send a frame to the broadcast address on the network segment and asking the computer with a specific IP to respond by providing its MAC).
- ICMP from Network Layer (has many types; ICMP ping uses Type 8 echo and Type 0 echo reply).
- If you want to ping a system on the same network, do an ARP query before ICMP Echo.
- TCP and UDP from Transport Layer (this method is efficient, especially when ICMP Echo is blocked).

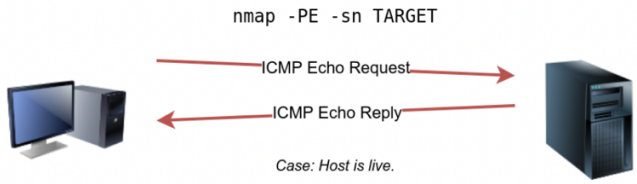




Nmap Host Discovery Using ARP:

- When an *unprivileged* user tries to scan targets outside the local network, Nmap reports to a TCP 3-way handshake by sending SYN packets to port 80 and 443.
- When a *privileged* user tries to scan targets outside the local network, Nmap uses ICMP echo requests, TCP ACK to port 80, TCP SYN to port 443, and ICMP timestamp request.
- ARP scan is possible only if you are on the same subnet as the target systems.

Nmap Host Discovery Using ICMP:



- In general, we don't expect to learn the MAC addresses of the targets unless they are on the same subnet as our system.
- Because ICMP echo requests tend to be blocked, you might also consider ICMP Timestamp or ICMP Address Mark requests to tell if a system is online.
- If one type of packet is being blocked, we can always choose another to discover the target network and services.
- To tell Nmap to use ICP Timestamp to discover live hosts we can use the option *-PP*.
- To tell Nmap to use ICMP Address Mask to discover live hosts we can use the option *-PM*.
- To tell Nmap to use ICMP Echo to discover life hosts we can use the option *-PE*.

Nmap Host Discovery Using TCP and UDP:

- We can conduct a TCP SYN Ping by sending a packet with the SYN flag set to a TCP port (80) and wait for a response. Depending on whether the port replies with a SYN.ACK or a RST (reset) we can infer whether the host is up.
- Root users can send TCP SYN packets that don't need to complete the TCP 3-way handshake even if the port is open, but unprivileged users have no choice but to complete the 3-way handshake if the port is open.

```
pentester@TryHackMe$ sudo nmap -PS -sn 10.10.68.220/24
Starting Nmap 7.92 ( https://nmap.org ) at 2021-09-02
13:45 EEST
Nmap scan report for 10.10.68.52
Host is up (0.10s latency).
Nmap scan report for 10.10.68.121
Host is up (0.16s latency).
Nmap scan report for 10.10.68.125
Host is up (0.089s latency).
Nmap scan report for 10.10.68.134
Host is up (0.13s latency).
Nmap scan report for 10.10.68.220
Host is up (0.11s latency).
Nmap done: 256 IP addresses (5 hosts up) scanned in 17.38
seconds
```

- TCP ACK Ping is also an option – sends a packet with an ACK flag set but you must be running Nmap as a privileged user to be able to accomplish this, otherwise Nmap will attempt a 3-way handshake.
- UDP Ping uses UDP to discover if the host is online. Sending a UDP packet to an open port is not expected to lead to any reply, but if we send a UDP packet to a closed UDP port, we expect to get an ICMP port unreachable packet which indicates that the target system is up and available.

```
pentester@TryHackMe$ sudo nmap -PU -sn 10.10.68.220/24
Starting Nmap 7.92 ( https://nmap.org ) at 2021-09-02
13:45 EEST
Nmap scan report for 10.10.68.52
Host is up (0.10s latency).
Nmap scan report for 10.10.68.121
Host is up (0.10s latency).
Nmap scan report for 10.10.68.125
Host is up (0.14s latency).
Nmap scan report for 10.10.68.134
Host is up (0.096s latency).
Nmap scan report for 10.10.68.220
Host is up (0.11s latency).
Nmap done: 256 IP addresses (5 hosts up) scanned in 9.20
seconds
```

- Masscan uses a similar approach to discover available systems but is aggressive with the rate of packets it generates.
- To run a TCP SYN ping scan on the telnet port using Nmap you can modify the *-p* option to be *-ps23*.

Using Reverse-DNS Lookup:

- Nmap's default behavior is to use reverse-DNS online hosts because the hostnames can reveal a lot, but you can use *-n* to skip this step.
- You can use option *-R* to query the DNS server even for offline hosts (by default will look up online hosts).

Summary:

By completing this lesson, I was able to take a deep dive into the Nmap service and the various functionality it provides. I learned how to perform ARP scans, ICMP scans, and TCP/UDP scans, and when to use each in order to gather the best information in the specific circumstance.

Scan Type	Example Command
ARP Scan	`sudo nmap -PR -sn MACHINE_IP/24`
ICMP Echo Scan	`sudo nmap -PE -sn MACHINE_IP/24`
ICMP Timestamp Scan	`sudo nmap -PP -sn MACHINE_IP/24`
ICMP Address Mask Scan	`sudo nmap -PM -sn MACHINE_IP/24`
TCP SYN Ping Scan	`sudo nmap -PS22,80,443 -sn MACHINE_IP/30`
TCP ACK Ping Scan	`sudo nmap -PA22,80,443 -sn MACHINE_IP/30`
UDP Ping Scan	`sudo nmap -PU53,161,162 -sn MACHINE_IP/30`