Nmap Notes

- Nmap, or Network Mapper, is an open source Linux command line tool for network exploration and security auditing.
- Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.
- With Nmap, server administrators can quickly reveal hosts and services, search for security issues, and scan for open ports.
- Open ports on a server are a security vulnerability that can potentially allow a hacker to exploit services on your network. If those services are unpatched, a hacker can easily take advantage of the system

- There are a large number of ports. 65,535 ports to be exact. This large number of ports can seem rather overwhelming when thinking about securing them individually, however, using the principal of least privilege you just close all ports and focus on opening the small number of ports you really need.

- Ports
    - Close All Ports!
    - Patch All Services
    - Incorporate Principal of Least Privilege
    - Document Open Ports

- The Nmap tool can audit and discover local and remote open ports, as well as network information and hosts.

- 6 port states
    - Open
        - An application is actively accepting TCP connections, UDP datagrams or SCTP associations on this port.
    - Closed
        - A closed port is accessible (it receives and responds to Nmap probe packets), but there is no application listening on it.
    - filtered
        - Nmap cannot determine whether the port is open because packet filtering prevents its probes from reaching the port. The filtering could be from a dedicated firewall device, router rules, or host-based firewall software. These ports frustrate attackers because they provide so little information. Sometimes they respond with ICMP error messages such as type 3 code 13 (destination unreachable: communication administratively prohibited), but filters that simply drop probes without responding are far more common. This forces Nmap to retry several times just in case the probe was dropped due to network congestion rather than filtering. This slows down the scan dramatically.
    - Unfiltered
        - The unfiltered state means that a port is accessible, but Nmap is unable to determine whether it is open or closed. Only the ACK scan, which is used to map firewall rulesets, classifies ports into this state. Scanning unfiltered ports with other scan types such as Window scan, SYN scan, or FIN scan, may help resolve whether the port is open
    - Open / Filtered
        - Nmap places ports in this state when it is unable to determine whether a port is open or filtered. This occurs for scan types in which open ports give no response.
    - Close Filtered
        - This state is used when Nmap is unable to determine whether a port is closed or filtered. It is only used for the IP ID idle scan.

1. Nmap Command to Scan for Open Ports
    - nmap subdomain.server.com
    - Without flags, as written above, Nmap reveals open services and ports on the given host or hosts.
    - -F “Fast Scan” flag does not scan as many ports, it isn’t as thorough.

2. Scan Multiple Hosts
    - nmap 192.168.0.1 192.168.0.2 192.168.0.3
    - nmap 192.168.0.* // Use the * wildcard to scan an entire subnet at once.
    - nmap 192.168.0.1,2,3
    - nmap 192.168.0.1-4

3. Excluding Hosts from Search
    - nmap 192.168.0.* --exclude 192.168.0.2
    - nmap 192.168.0.* --excludefile /file.txt

4. Scan to Find out OS Information
    - Nmap can also provide operating system detection, script scanning, traceroute, and version detection. It’s important to note that Nmap will do its best to identify things like operating systems and versions, but it may not always be entirely accurate.
    - nmap -A 192.168.0.1
    - Using the -O flag on your Nmap command will reveal further operating system information of the mapped hosts. The -O flag enables OS detection. Additional tags include –osscan-limit and -osscan-guess.
    - nmap -O 192.168.0.1
    - The “–osscan-limit” command will only guess easy operating system targets. The “-osscan-guess” command will be more aggressive about guessing operating systems. Again, operating systems are detected based on certain hallmarks: it isn’t a certainty that the information is accurate.

5. Scan to Detect Firewall Settings
    - nmap -sA 192.168.0.1

6. Find Information About Service Versions
    - nmap -sV 192.168.0.1
    - At times, you may need to detect service and version information from open ports. This is useful for troubleshooting, scanning for vulnerabilities, or locating services that need to be updated.
    - You can use –version-intensity “level” from 0 to 9 to determine the intensity level of this search. You can also use –version-trace to show more detailed information of the scan if the scan does not come out with the results that you would ordinarily expect

7. Port Scanning
    - nmap -p 443 192.168.0.1
    - nmap -p T:8888,443 192.168.0.1 // By adding a type of port before the port itself, you can scan for information regarding a specific type of connection
    - nmap -p 80,443 192.168.0.1

- You can scan for multiple ports with the -p flag by separating them with a comma.
- nmap -p 80-443 192.168.0.1
- You can also scan for multiple ports with the -p flag by marking a range with the hyphen. To scan ports in order rather than randomly, add the flag "-r" to the command.
- You can also use the command "–top-ports" followed by a number to find the most common ports, up to that amount.

8. Complete a Scan in Stealth Mode
   - Using the "-sS" flag will initiate a stealth scan with TCP SYN.
   - nmap -sS 192.168.0.1
9. Identify Hostnames
   - nmap -sL 192.168.0.1
10. Scan from a File
    - nmap -iL /file.txt
11. Get More Information with Verbose
    - nmap -v 192.168.0.1
    - Verbose output generally gives you far more information regarding a command.
    - Very helpful in the case of debugging
12. Scan IPv6 Addresses
    - nmap -6 ::ffff:c0a8:1
13. Scan to See Which Servers are Active
    - nmap -sP 192.168.0.0/24
14. Find Host Interfaces, Routes, and Packets
    - nmap --iflist // command will produce a list of the relevant interfaces and routes.
    - nmap --packet-trace // "–packet-trace" will show packets sent and received, providing similar value for debugging.
15. Aggressive Scans and Timings
    - nmap -T5 192.168.0.1
    - In NMAP, timing controls both the speed and the depth of the scan.
    - An aggressive scan is going to be faster, but it also could be more disruptive and also inaccurate. There are other options such as T1, T2, T3, and T4 scans. For most scans, T3 and T4 timings will be sufficient.
16. Help
    - nmap -h
17. Create Decoys While Scanning
    - Nmap can also be used to create decoys, which are intended to fool firewalls. While decoys can be used for nefarious purposes, it's generally used to debug.
    - nmap -D 192.168.0.1,192.168.0.2,...