scanning

Keiichi Shima <keiichi@iijlab.net>

Slides stolen from

Matsuzaki ‘maz’ Yoshinobu <maz@iij.ad.jp>

nmap

- nmap https://nmap.org/
- nmap is a free and open source network exploration and security auditing tool
- nmap was created by Gordon Lyon, a.k.a. Fyodor Vaskovich, and first published in 1997
- Working cross-platform although best working on Linux-type environments
- It uses raw IP packet to determine
  - What hosts are available on the network
  - What services (application name and version)
  - Guesses the operating system, uptime and other characteristics

Ethical Issues

- Can be used for hacking to discover vulnerable servers
- System admins can use it to check that their systems meet security requirements
- Unauthorized use of nmap on a foreign system could be illegal
- Make sure if you have permission to nmap before using this tool
- Do not abuse nmap, do not excuse

How it works

- DNS lookup-matches name with IP
- Try to call connect() system call to check if target ports open
- Try to send TCP SYN and wait for SYN/ACK
- Try to send a UDP packet and wait for ICMP PORT UNREACH
- Many artistic mechanisms to scan hosts are implemented
  - See more info at https://nmap.org/nmap_doc.html

Scanning Techniques

- Host discovery with in a specific subnet
- Port scanning specifying protocol types, port ranges
- OS detection
- Service version detection
- Scripting engine support for advanced scanning
- Scan timing and performance control
- Firewall, IDS evasion, and spoofing
- Report the result

Scan Options

```
Nmap 7.70 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -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: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports consecutively - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>
```

Scan Options

```
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
```

Scan Options

```
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
```

Scan Options

```
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)

OS DETECTION:
  -O: Enable OS detection
  --osscan-limit: Limit OS detection to promising targets
  --osscan-guess: Guess OS more aggressively
```

Scan Options

```
MISC:
  -6: Enable IPv6 scanning
  -A: Enable OS detection, version detection, script scanning, and
traceroute
```

Hands on