

# Cyber Threat Emulation (CTE)

## Module 2, Lesson 2: Active Scanning and Enumeration

## Course Objectives

After completing this course, students will be able to:

- Summarize the CTE squad's responsibilities, objectives, and deliverables from each CPT stage
- Analyze threat information
- Develop a Threat Emulation Plan (TEP)
- Generate mitigative and preemptive recommendations for local defenders
- Develop mission reporting
- Conduct participative operations
- Conduct reconnaissance
- Analyze network logs for offensive and defensive measures

## Course Objectives (Continued)

Students will also be able to:

- Analyze network traffic and tunneling protocols for offensive and defensive measures
- Plan non-participative operations using commonly used tools, techniques and procedures (TTPs)

## Module 2: Threat Emulation (Objectives)

- Conduct reconnaissance
- Generate mission reports from non-participative operations
- Plan a non-participative operation using social engineering
- Plan a non-participative operation using Metasploit
- Analyze network logs for offensive and defensive measures
- Analyze network traffic and tunneling protocols for offensive and defensive measures
- Plan a non-participative operation using Python
- Develop fuzzing scripts
- Develop buffer overflow exploits

## Module 2 – Lesson 2: Active Scanning and Enumeration (Objectives)

- Conduct active reconnaissance
- Develop mission reports from results of exploitation

## Methods of Scanning

### Passive discovery techniques

- Monitor communications
- Transparent
- Take more time

### Active discovery techniques

- Fast
- Provide a lot of information
- Can trigger alerts

## Methods of Scanning

### Port Scanning

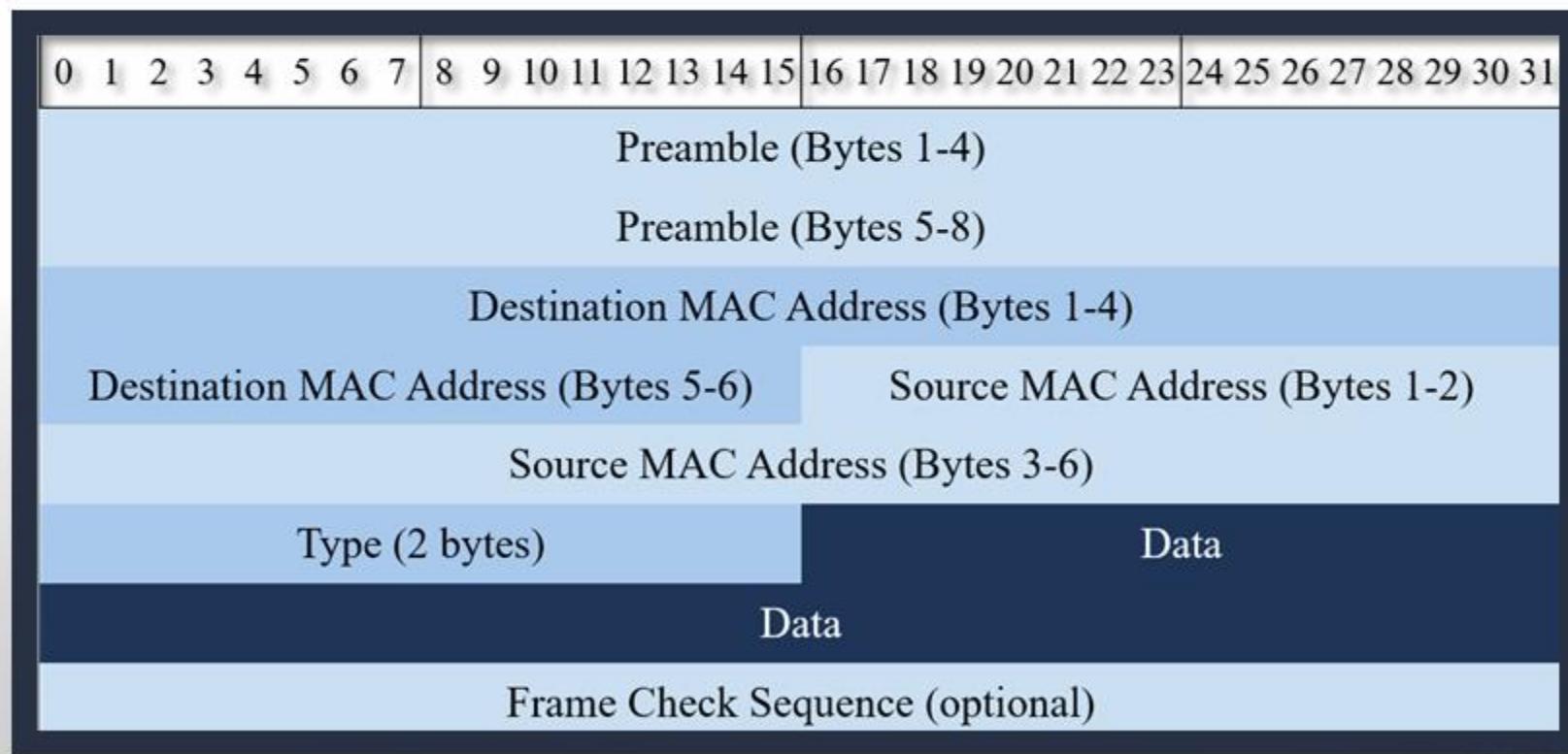
- Determining ports that are open
- Reveals presence of devices
- Reconnaissance tool for attackers

### Vulnerability Scanning

- Combines port scanning
- Reveals hosts and servers for known vulnerabilities
- Provides report

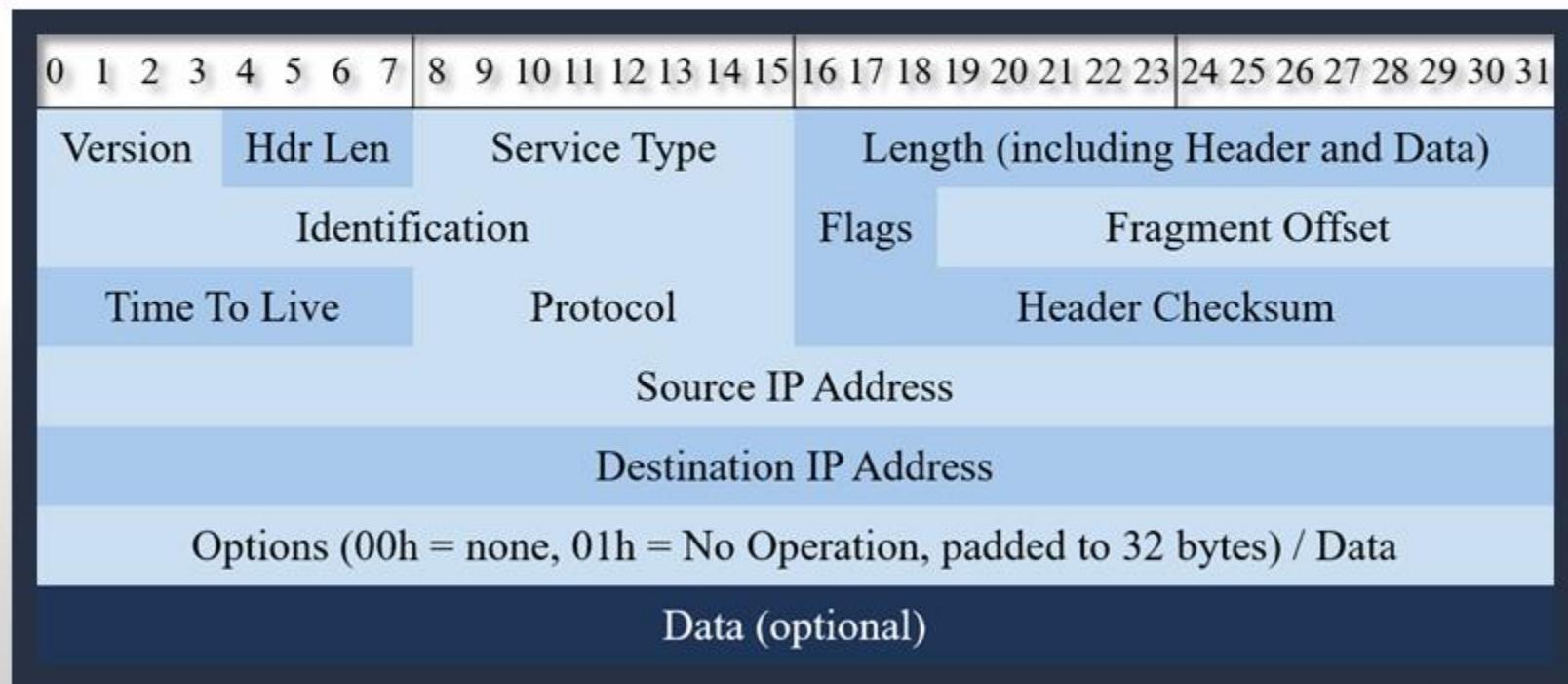
# Major Protocols Review

- Ethernet



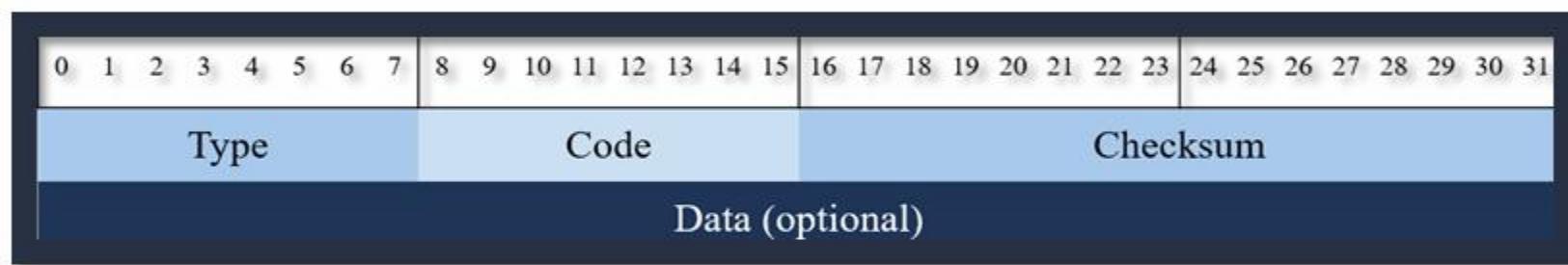
# Major Protocols Review

- IPv4



# Major Protocols Review

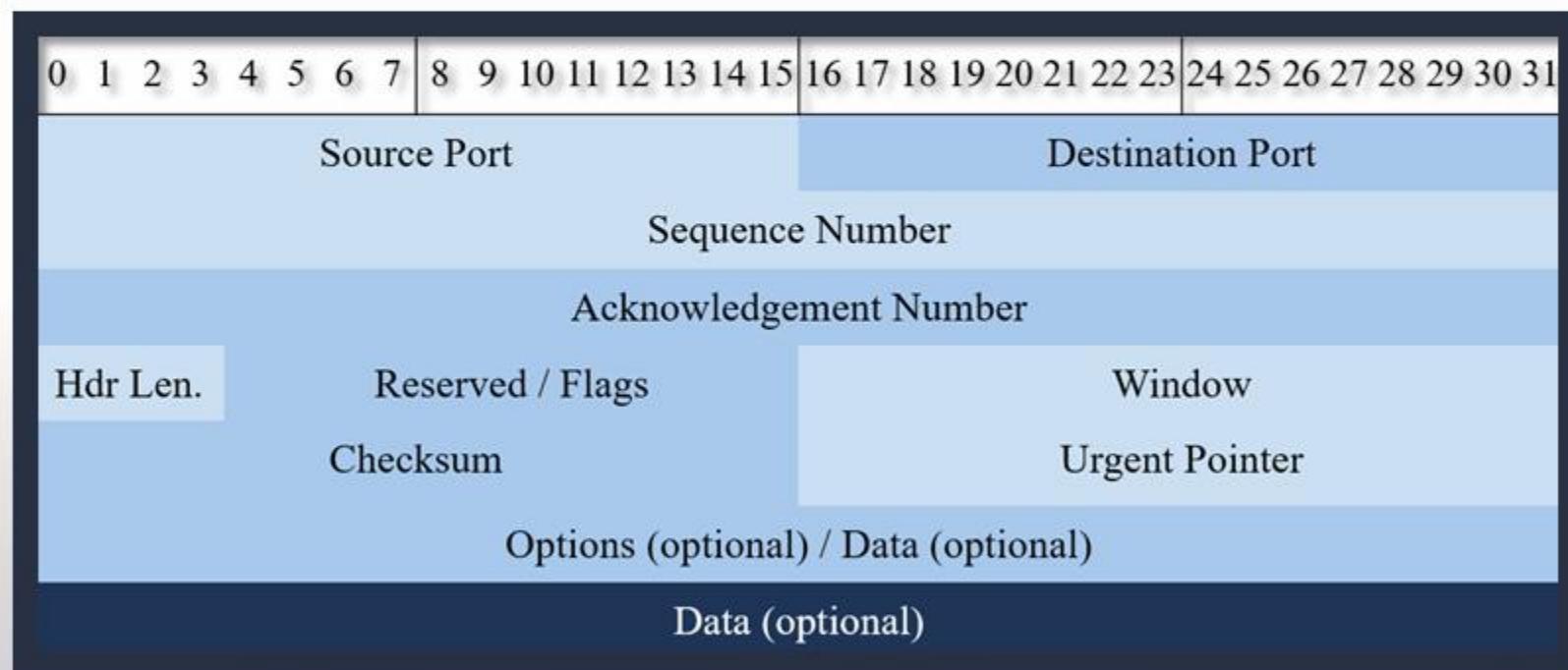
- ICMP



ICMP Type	Purpose
0x08	ICMP Echo Request
0x00	ICMP Echo Reply
0x03	ICMP Destination Unreachable
0x0B	ICMP TTL Exceeded

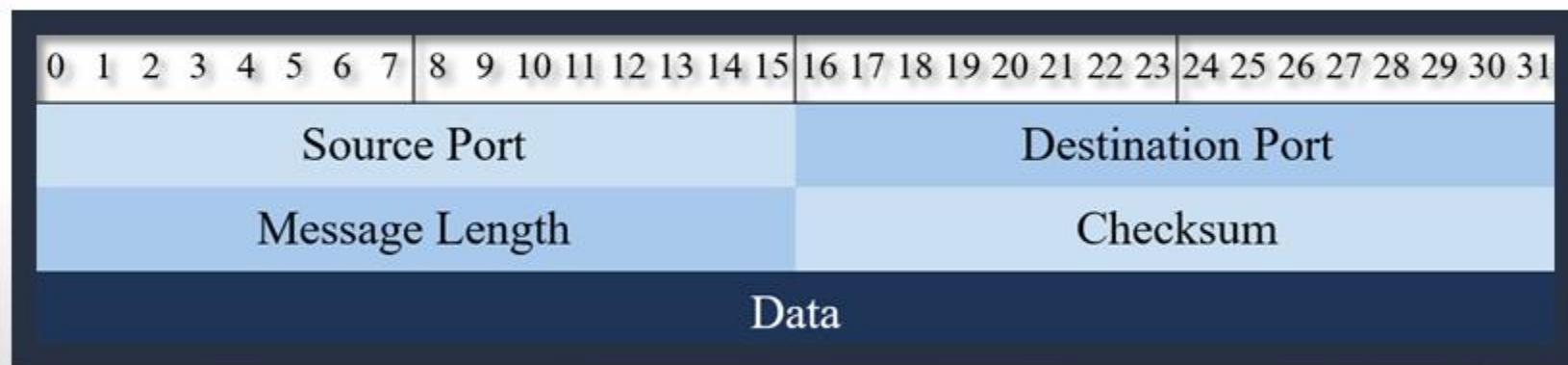
# Major Protocols Review

- TCP



## Major Protocols Review

- UDP



## Active Scanning Techniques

- Discovering Hosts (Network Mapping)
  - MAC and IP addresses
  - Host names
  - Operating systems (OSs)
  - Services running
- Broadcast pings and ping sweeps
- ARP scans
- ICMPv6 neighbor discovery

## Active Scanning Techniques

- Scanning ports
  - What will happen when connecting to a TCP port?
  - What about UDP?
- OS detection
- Service and version detection
- Timing and optimization
- Firewall and IDS evasion
- Packet manipulation

## Scapy

- What is Scapy?
- Why use Scapy?
- Important concepts
- Crafting packets
- Sending and receiving packets

## What is Scapy?

- Program for manipulating packets
- Capable of sniffing and transmitting packets
- Can handle many tasks:
  - Scanning
  - Traceroute
  - Host discovery
  - Probing
  - And more



## Scapy

- Very useful tool
- Cross platform
- Scripting in Python
- Replay packets



## Crafting Packets with Scapy

- Create a packet
- Show the contents of the packet

```
>>> i=IP()  
>>> i.src="192.168.229.55"  
>>> i.dst="192.168.229.13"  
>>> icmp=ICMP()  
>>> icmp.type=8  
>>> icmp.code=0
```

```
>>> packet.show()
```

- Combine the layers

```
>>> packet=i/icmp
```

## Sending Packets with Scapy

- Sending packets is easy!

```
>>> send(packet)
.
Sent 1 packets.
>>> █
```

- Or you could use

```
>>> send(IP(src="192.168.229.55", dst="192.168.229.13")/ICMP(type=8, code=0))
.
Sent 1 packets.
```

# Sending and Receiving Packets

## Functions:

- Send and Receive
  - sr()
- Send and Receive one packet
  - sr1()

```
>>> sr1(IP(dst="192.168.229.13")/ICMP()/ "Hello World")
Begin emission:
.Finished sending 1 packets.
*
Received 2 packets, got 1 answers, remaining 0 packets
<IP version=4 ihl=5 tos=0x0 len=39 id=102 flags= frag=0 ttl=128 proto=icmp cksum=0xeef3 src=192.168.229.13 dst=192.168.229.29 options=[] |<ICMP type=echo-reply code=0 cksum=0xae31 id=0x0 seq=0x0 |<Raw load='Hello World' |<Padding load='\x00\x00\x00\x00\x00\x00\x00\x00' |>>>
```

# Nmap

Features include

- Host and port scanning
- OS detection
- Detecting versions
- Scriptable

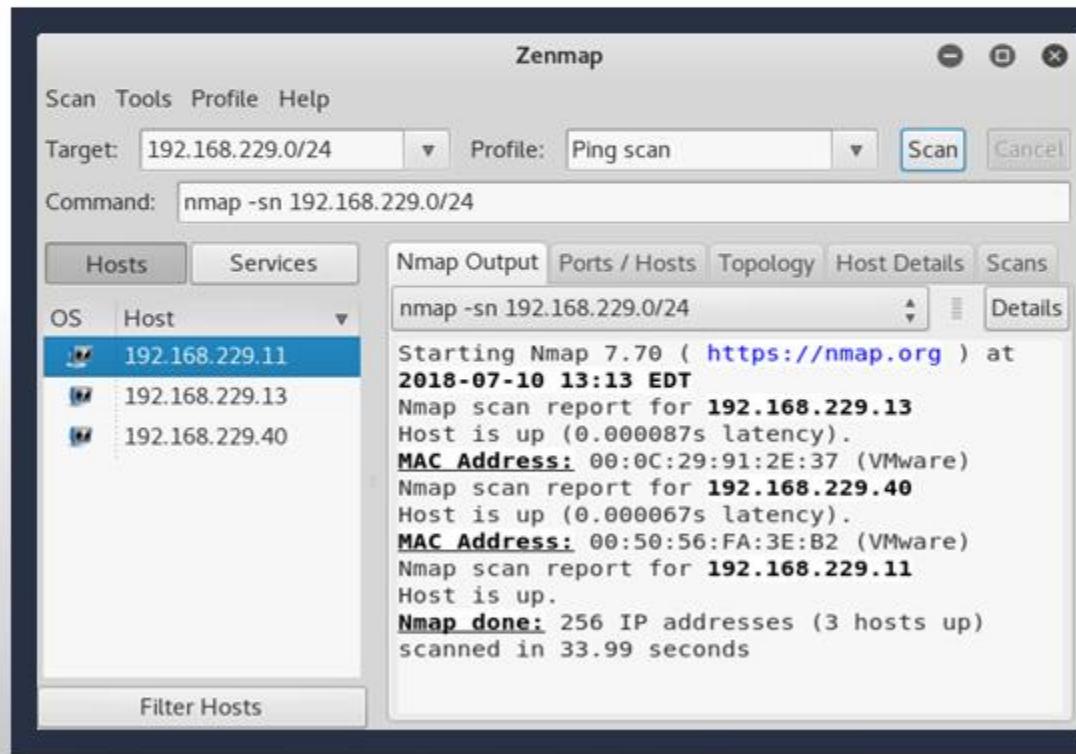
Uses include

- Mapping networks
- Identifying open ports
- Security Auditing



# Nmap

Graphical User Interface (Zenmap):



Command Line:

```
root@kali:~# nmap -sn 192.168.229.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2018-07-25 10:21 EDT
Nmap scan report for 192.168.229.13
Host is up (0.00017s latency).
MAC Address: 00:0C:29:E8:6F:4E (VMware)
Nmap scan report for 192.168.229.40
Host is up (0.000050s latency).
MAC Address: 00:50:56:FA:3E:B2 (VMware)
Nmap scan report for 192.168.229.29
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 33.84 seconds
root@kali:~#
```

# Nmap Options

```
root@kali:~# nmap
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
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
```

# Mapping the Environment – Discovering Hosts

## Discovery options

- List scan (-sL)
- No port scan (-sn)
- No ping (-Pn)
- TCP SYN ping (-PS)
- TCP ACK ping (-PA)
- UDP ping (-PU)
- SCTP INIT ping (-PY)
- ICMP ping Types (-PE, -PP, -PM)
- IP protocol ping (-PO)
- ARP ping (-PR)

## Mapping the Environment - Outputting Results

### OUTPUT:

- oN/-oX/-oS/-oG <file>: Output scan in normal, XML, s|<rIpt kIddi3, and Grepable format, respectively, to the given filename.
- oA <basename>: Output in the three major formats at once
- v: Increase verbosity level (use -vv or more for greater effect)
- d: Increase debugging level (use -dd or more for greater effect)
- reason: Display the reason a port is in a particular state
- open: Only show open (or possibly open) ports
- packet-trace: Show all packets sent and received
- iflist: Print host interfaces and routes (for debugging)
- append-output: Append to rather than clobber specified output files
- resume <filename>: Resume an aborted scan
- stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML
- webxml: Reference stylesheet from Nmap.Org for more portable XML
- no-stylesheet: Prevent associating of XSL stylesheet w/XML output

## Mapping the Environment – Port Scanning

Option	Scan Type
-sP	Ping scan only
-sS	TCP SYN
-sT	TCP connect
-sA	TCP ACK
-sN / -sF / -sX	TCP NULL, FIN, and Xmas
-sW	TCP window
-sM	TCP Miamon
-sU	UDP
-sI	Idle
-b	FTP bounce
-sO	IP protocol

# OS Fingerprinting and Version Detection

## OS DETECTION:

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

## 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)

## Nmap Scripting Engine (NSE)

Automates networking tasks

- Network discovery
- More sophisticated version detection
- Vulnerability detection
- Backdoor detection
- Vulnerability exploitation

# Nmap Scripting Engine (NSE)

## Types of Scripts:

- Prerule
- Host
- Service
- Postrule

## Categories:

- |             |             |
|-------------|-------------|
| • Auth      | • Fuzzer    |
| • Broadcast | • Intrusive |
| • Brute     | • Malware   |
| • Default   | • Safe      |
| • Discovery | • Version   |
| • DOS       | • Vuln      |
| • Exploit   |             |
| • External  |             |

# Using Nmap Scripting Engine (NSE)

- Selecting scripts
  - locate \*.nse
  - find / -name “\*.nse”
- Performing Script Scans

```
SCRIPT SCAN:  
  -sC: equivalent to --script=default  
  --script=<Lua scripts>: <Lua scripts> is a comma separated list of  
    directories, script-files or script-categories  
  --script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts  
  --script-args-file=filename: provide NSE script args in a file  
  --script-trace: Show all data sent and received  
  --script-updatedb: Update the script database.  
  --script-help=<Lua scripts>: Show help about scripts.  
    <Lua scripts> is a comma-separated list of script-files or  
    script-categories.
```

# Ncat

```
root@kali:~# ncat -h
Ncat 7.70 ( https://nmap.org/ncat )
Usage: ncat [options] [hostname] [port]

Options taking a time assume seconds. Append 'ms' for milliseconds,
's' for seconds, 'm' for minutes, or 'h' for hours (e.g. 500ms).
-4                                Use IPv4 only
-6                                Use IPv6 only
-U, --unixsock                      Use Unix domain sockets only
-C, --crlf                           Use CRLF for EOL sequence
-c, --sh-exec <command>             Executes the given command via /bin/sh
-e, --exec <command>                Executes the given command
--lua-exec <filename>               Executes the given Lua script
-g hop1[,hop2,...]                  Loose source routing hop points (8 max)
-G <n>                             Loose source routing hop pointer (4 8 12 ...)
```

# Nping

```
root@kali:~# nping
Nping 0.7.70 ( https://nmap.org/nping )
Usage: nping [Probe mode] [Options] {target specification}

TARGET SPECIFICATION:
Targets may be specified as hostnames, IP addresses, networks, etc.
Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.*.1-24

PROBE MODES:
--tcp-connect          : Unprivileged TCP connect probe mode.
--tcp                  : TCP probe mode.
--udp                  : UDP probe mode.
--icmp                : ICMP probe mode.
--arp                  : ARP/RARP probe mode.
--tr, --traceroute    : Traceroute mode (can only be used with
                       : TCP/UDP/ICMP modes).

TCP CONNECT MODE:
-p, --dest-port <port spec>   : Set destination port(s).
-g, --source-port <portnumber> : Try to use a custom source port.

TCP PROBE MODE:
-a, --source-port <portnumber> : Set source port.
```

## Nping

- Use `man nping` to find syntax examples

A typical Nping execution is shown in Example 1. The only Nping arguments used in this example are `-c`, to specify the number of times to target each host, `--tcp` to specify TCP Probe Mode, `-p 80,433` to specify the target ports; and then the two target hostnames.

### Example 1. A representative Nping execution

```
# nping -c 1 --tcp -p 80,433 scanme.nmap.org google.com
```

```
Starting Nping ( https://nmap.org/nping )
```

```
SENT (0.0120s) TCP 96.16.226.135:50091 > 64.13.134.52:80 S ttl=64 id=52072 iplen=40 seq=1077657
```

```
88 win=1480
```

# Nping

```
root@kali:~# nping -c 1 --tcp -p 80,433 192.168.229.13 192.168.229.80

Starting Nping 0.7.70 ( https://nmap.org/nping ) at 2018-07-25 13:50 EDT
SENT (0.0818s) TCP 192.168.229.29:1837 > 192.168.229.13:80 S ttl=64 id=4814 iplen=40 seq=145641095 win=1480
SENT (1.0827s) TCP 192.168.229.29:1837 > 192.168.229.80:80 S ttl=64 id=4814 iplen=40 seq=145641095 win=1480
RCVD (1.0834s) TCP 192.168.229.80:80 > 192.168.229.29:1837 SA ttl=128 id=891 iplen=44 seq=3928731479 win=8192 <mss 1460>
SENT (2.0840s) TCP 192.168.229.29:1837 > 192.168.229.13:433 S ttl=64 id=4814 iplen=40 seq=145641095 win=1480
SENT (3.0857s) TCP 192.168.229.29:1837 > 192.168.229.80:433 S ttl=64 id=4814 iplen=40 seq=145641095 win=1480
RCVD (3.0860s) TCP 192.168.229.80:433 > 192.168.229.29:1837 RA ttl=128 id=892 iplen=40 seq=0 win=0

Statistics for host 192.168.229.13:
| Probes Sent: 2 | Rcvd: 0 | Lost: 2 (100.00%)
|_ Max rtt: N/A | Min rtt: N/A | Avg rtt: N/A
Statistics for host 192.168.229.80:
| Probes Sent: 2 | Rcvd: 2 | Lost: 0 (0.00%)
|_ Max rtt: 0.567ms | Min rtt: 0.177ms | Avg rtt: 0.372ms
Raw packets sent: 4 (160B) | Rcvd: 2 (92B) | Lost: 2 (50.00%)
Nping done: 2 IP addresses pinged in 3.12 seconds
```

## Ndiff

```
root@kali:~# ndiff scan1.xml scan2.xml
-Nmap 7.70 scan initiated Mon Jul 16 09:12:34 2018 as: nmap -sS -oX scan1.xml 19
2.168.229.13
+Nmap 7.70 scan initiated Mon Jul 16 08:40:37 2018 as: nmap -sS -oX scan2.xml 19
2.168.229.13

192.168.229.13, 00:0C:29:91:2E:37:
-Not shown: 990 closed ports
+Not shown: 989 closed ports
  PORT      STATE SERVICE VERSION
+80/tcp    open  http
root@kali:~#
```

## Evasion Techniques

- Customizing TCP scan flag (--scan-flags)
- Fragmentation (-f / --mtu)
- Adding decoy IP addresses (-D)
- Idle Scan (-sl)
- Changing the Source port (-g)
- Spoofing IP (-S) and MAC address (--spoof-mac)
- Randomize target scan order (--randomize-hosts)
- Add random data to packets (--data-length)
- Manipulating the time-to-live field (--ttl)
- Send packets with bogus TCP or UDP checksums (--badsums)
- Firewalk (--script firewalk)

# Evasion Techniques

```
root@kali:~# nmap 192.168.229.80 -p 80 -sC
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-26 21:24 EEST
Nmap scan report for 192.168.229.80
Host is up (0.00031s latency).

PORT      STATE SERVICE
80/tcp      open  http
| http-methods:
|_ Potentially risky methods found
|_ http-title: IIS7
MAC Address: 00:0C:29 (Microsoft Corporation)

Nmap done: 1 IP address scanned in 0.04s
```

ip.addr==192.168.229.80

No.	Time	Source	Destination	Protocol	Length	
	41	5.072546990	192.168.229.29	192.168.229.80	TCP	6
→	42	5.072936370	192.168.229.29	192.168.229.80	HTTP	37
	43	5.072990485	192.168.229.29	192.168.229.80	HTTP	21
	44	5.073036035	192.168.229.29	192.168.229.80	TCP	9

Connection: close\r\nUser-Agent: Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/\r\nHost: 192.168.229.80\r\nContent-Length: 88\r\nContent-Type: application/x-www-form-urlencoded\r\nContent-Length: 88\r\n\r\n0060 6c 6f 73 65 0d 0a 55 73 65 72 2d 41 67 65 6e 74 lose..User-Agent
0070 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28 63 : Mozilla/5.0 (c
0080 6f 6d 70 61 74 69 62 6c 65 3b 20 4e 6d 61 70 20 ompatible; Nmap
0090 53 63 72 69 70 74 69 6e 67 20 45 6e 67 69 6e 65 Scripting Engine
00a0 3b 20 68 74 74 70 73 3a 2f 2f 6e 6d 61 70 2e 6f ; https://nmap.o
00b0 72 67 2f 62 6f 6f 6b 2f 6e 73 65 2e 68 74 6d 6c rg/book/nse.html
00c0 29 0d 0a 48 6f 73 74 3a 20 31 39 32 2e 31 36 38 )..Host: 192.168
00d0 2e 32 32 39 2e 38 30 0d 0a 43 6f 6e 74 65 6e 74 .229.80. .Content

# Evasion Techniques

```
root@kali:~# nmap 192.168.229.80 -p 80 -sC -script-args http.useragent="Some other Browser"
Starting Nmap 7.1 ( https://nmap.org ) at 2017-05-22 13:45 CEST
Nmap scan report for 192.168.229.80
Host is up (0.000s latency).
PORT      STATE SERVICE
80/tcp    open  http
| http-methods:
|_ Potentially dangerous methods found
|_ http-title: IIS 8.0
MAC Address: 00:0C:29 (Microsoft Corporation)
Nmap done: 1 IP address scanned in 0.04s
```

ip.addr==192.168.229.80

No.	Time	Source	Destination	Protocol	Length
82	35.315968064	192.168.229.29	192.168.229.80	HTTP	18
83	35.316027774	192.168.229.29	192.168.229.80	HTTP	22
84	35.316087320	192.168.229.29	192.168.229.80	HTTP	17
85	35.316141196	192.168.229.29	192.168.229.80	TCP	9

Request Version: HTTP/1.1  
Connection: close\r\nHost: 192.168.229.80\r\nUser-Agent: Some other Browser\r\n

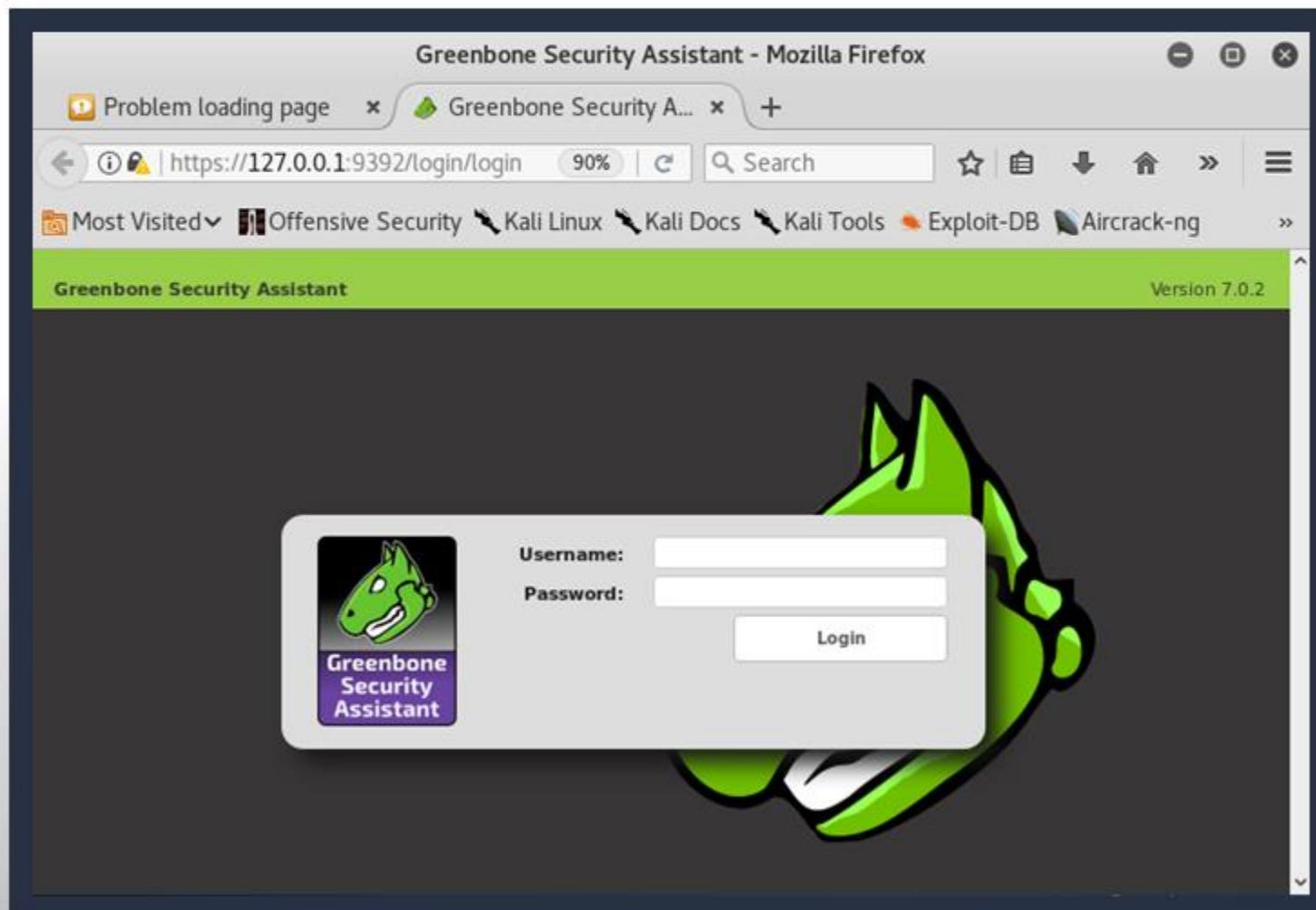
Hex	Dec	Text
0070	36 38 2e 32 32 39 2e 38 30 0d 0a 55 73 65 72 2d	68.229.8 0..User-
0080	41 67 65 6e 74 3a 20 53 6f 6d 65 20 6f 74 68 65	Agent: S ome othe
0090	72 20 42 72 6f 77 73 65 72 0d 0a 41 75 74 68 6f	r Browse r..Autho
00a0	72 69 7a 61 74 69 6f 6e 3a 20 4e 54 4c 4d 20 54	rization : NTLM T
00b0	6c 52 4d 54 56 4e 54 55 41 41 42 41 41 41 41 42	1RMTVNTU AABAAAAB
00c0	34 49 49 6f 41 41 41 41 41 41 41 41 41 41 41 41	4IIoAAAA AAAAAAAA

## OpenVAS

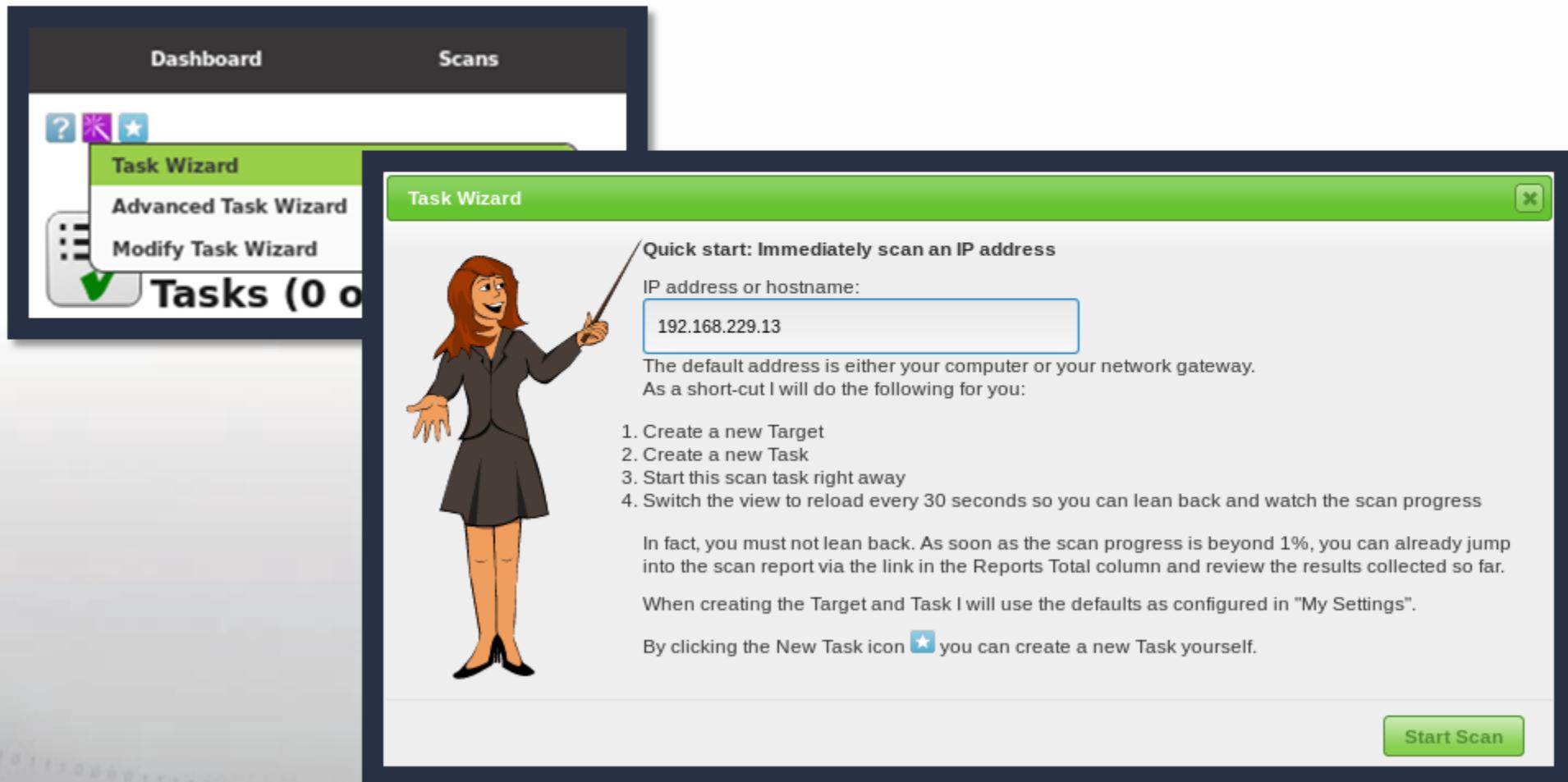
- Open source vulnerability scanner
- It's free!
- Over 50,000 network vulnerability tests
- Identifies vulnerabilities - Does not fix them
- Allows customized:
  - Targets
  - Scans
  - Depth of scans, and scan regions
  - Reports



# OpenVAS – Accessing the Web Interface



# OpenVAS – Configuring a Scan



# OpenVAS – Adding Targets

The screenshot shows the Greenbone Security Assistant web interface. The top navigation bar includes links for Dashboard, Scans, Assets, SecInfo, Configuration, and Targets. The Configuration menu is open, showing options like Targets, Port Lists, Credentials, Scan Configs, Alerts, Schedules, Report Formats, Agents, Scanners, Filters, Tags, and Permissions. The main content area displays a donut chart titled 'Tasks (1 of 1)' with one task in the 'N/A' category. Below the chart is a section titled 'Tasks with most Hi' which states 'No Tasks with High severity found'.

# OpenVAS – Adding Targets

New Target

Name: unnamed

Comment:

Hosts:

Manual: 127.0.0.1  
 From file: Browse... No file selected.  
 From host assets (0 hosts)

Exclude Hosts:

Reverse Lookup Only:  Yes  No

Reverse Lookup Unity:  Yes  No

Port List: All IANA assigned TCP 2...

Alive Test: Scan Config Default

Credentials for authenticated checks:

SSH:  on port 22

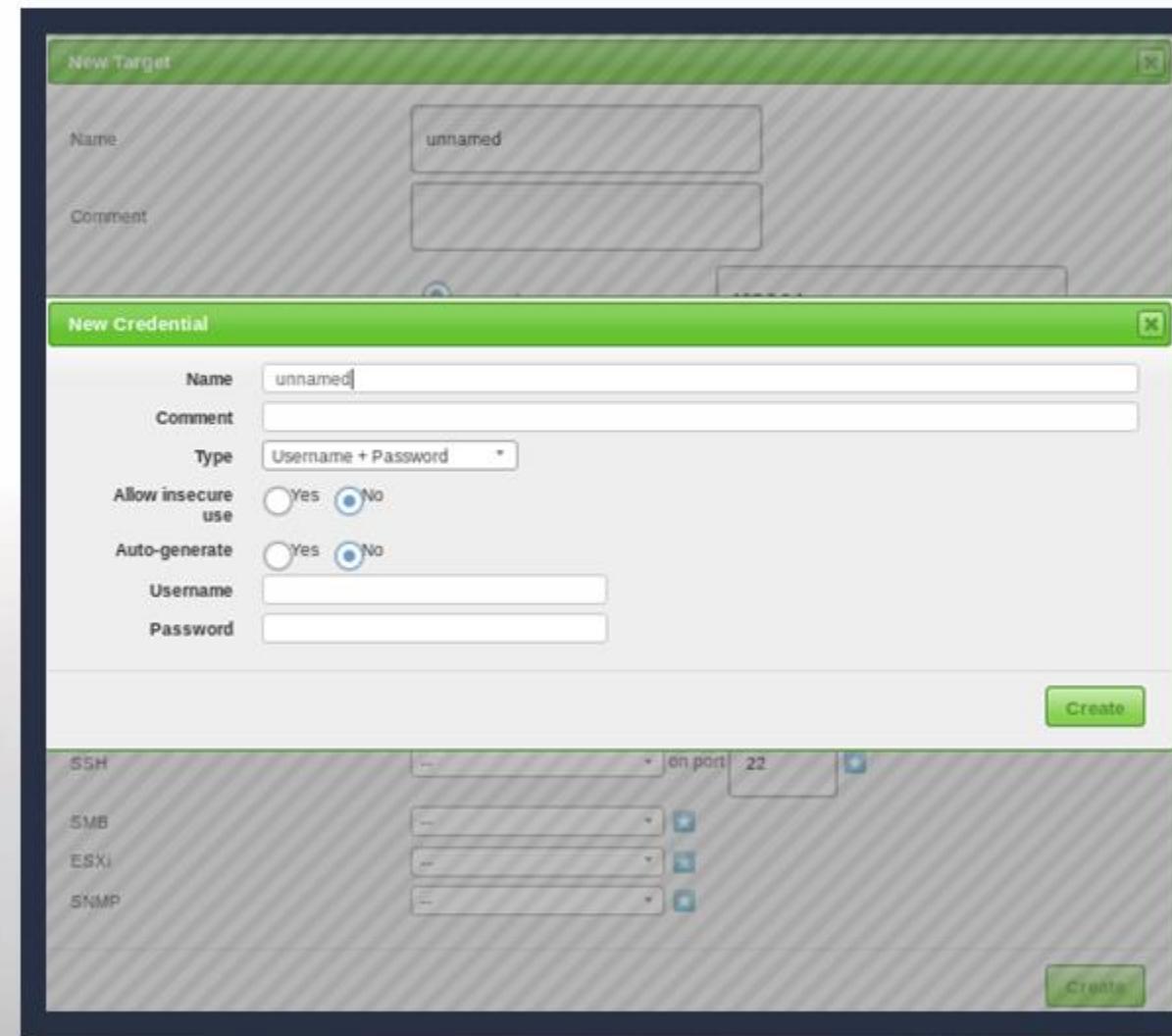
SMB:

ESXi:

SNMP:

**Create**

# OpenVAS – Adding Credentials



# OpenVAS – Changing the Network Vulnerability Tests (NVTs)

The screenshot shows the 'Scan Configs' page of the OpenVAS interface. The title bar says 'Scan Configs (8 of 8)'. Below is a table with the following data:

Name	Families		NVTs		Actions
	Total	Trend	Total	Trend	
<b>Discovery</b> (Network Discovery scan configuration.)	22	↗	2276	↗	
<b>empty</b> (Empty and static configuration template.)	0	↗	0	↗	
<b>Full and fast</b> (Most NVT's; optimized by using previously collected information.)	62	↗	46417	↗	
<b>Full and fast ultimate</b> (Most NVT's including those that can stop services/hosts; optimized by using previously collected information.)	62	↗	46417	↗	
<b>Full and very deep</b> (Most NVT's; don't trust previously collected information; slow.)	62	↗	46417	↗	
<b>Full and very deep ultimate</b> (Most NVT's including those that can stop services/hosts; don't trust previously collected information; slow.)	62	↗	46417	↗	
<b>Host Discovery</b> (Network Host Discovery scan configuration.)	2	↗	2	↗	
<b>System Discovery</b> (Network System Discovery scan configuration.)	6	↗	29	↗	

At the bottom left: '(Applied filter: rows=10 first=1 sort=name)'. At the bottom right: 'vApply to page contents' and page navigation controls.

# OpenVAS – Reviewing Scan Results

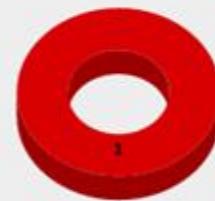
Greenbone Security Assistant

Refresh every 30 Sec. Logged in as: Admin admin | Logout Thu Jul 26 19:34:34 2018 UTC

Dashboard Scans Assets SecInfo Configuration Extras Administration Help

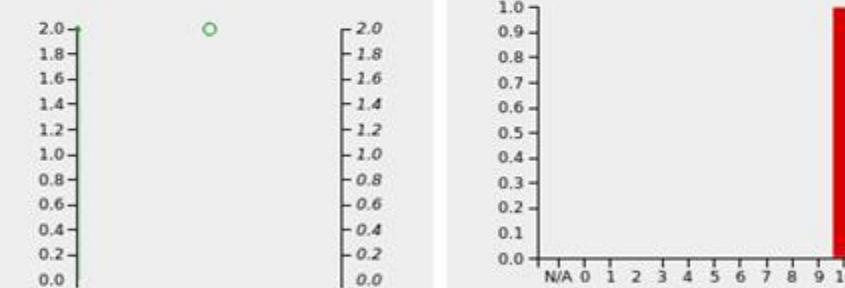
Reports by Severity Class (Total: 1)

High

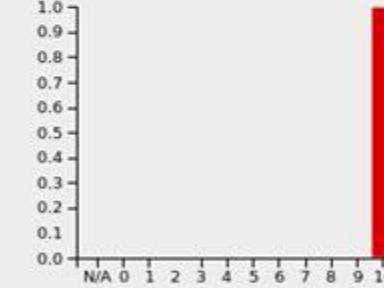


Reports: High results timeline

Max. High Max. High / host



Reports by CVSS (Total: 1)



Date Status Task Severity Scan Results Actions

Date	Status	Task	Severity	Scan Results	Actions
Thu Jul 26 18:30:37 2018	Done	Immediate scan of IP 192.168.229.80	10.0 (High)	High Medium Low Log False Pos.	 

(Applied filter: min\_qod=70 apply\_overrides=1 rows=10 sort-reverse=date first=1)

# OpenVAS – Reviewing Scan Results

**Greenbone Security Assistant**

Logged in as Admin admin | Logout  
Thu Jul 26 19:49:19 2018 UTC

Dashboard   Scans   Assets   SecInfo   Configuration   Extras   Administration   Help

PDF   Done

Filter:    [ ] [ ] [ ] [ ] [ ]

autofp=0 apply\_overrides=1 notes=1 overrides=1 result\_hosts\_only=1 first=1 rows=100 sort-reverse=severity levels=hml min\_qod=70

 Report: Results (5 of 24)

ID: 9fe22fe5-d234-4f38-9c95-ccb0b788d7c4  
Modified: Thu Jul 26 18:40:54 2018  
Created: Thu Jul 26 18:31:47 2018  
Owner: admin

1 - 5 of 5

Vulnerability	+	+	Severity	QoD	Host	Location	Actions
MS15-034 HTTP.sys Remote Code Execution Vulnerability (remote check)			10.0 (High)	95%	192.168.229.80	80/tcp	
Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)			9.3 (High)	95%	192.168.229.80	445/tcp	
DCE/RPC and MSRPC Services Enumeration Reporting			5.0 (Medium)	80%	192.168.229.80	135/tcp	
Microsoft IIS Default Welcome Page Information Disclosure Vulnerability			5.0 (Medium)	70%	192.168.229.80	80/tcp	
TCP timestamps			2.6 (Low)	80%	192.168.229.80	general/tcp	

(Applied filter:autofp=0 apply\_overrides=1 notes=1 overrides=1 result\_hosts\_only=1 first=1 rows=100 sort-reverse=severity levels=hml min\_qod=70)   1 - 5 of 5

Backend operation: 0.34s   Greenbone Security Assistant (GSA) Copyright 2009 - 2018 by Greenbone Networks GmbH, www.greenbone.net

# Nessus – Accessing

The screenshot shows the Nessus web interface. At the top, there is a dark header bar with the Nessus logo on the left, followed by 'Scans' and 'Settings' menu items, and a notification/bell icon and user profile icon on the right.

The main content area has a light gray background. On the left, there is a sidebar with a vertical list:

- FOLDERS**
  - My Scans** (highlighted with a green vertical bar)
  - All Scans
  - Trash
- RESOURCES**
  - Policies
  - Plugin Rules
  - Scanners

The main panel displays the 'My Scans' folder. It contains a message: "This folder is empty. [Create a new scan.](#)". At the top of this panel are three buttons: "Import", "New Folder", and a blue button labeled "New Scan" with a white plus sign.

# Nessus – Configuring a Scan

Scan Templates

[« Back to Scans](#)

Scanner

Search Library

 <b>Advanced Scan</b> Configure a scan without using any recommendations.	 <b>Audit Cloud Infrastructure</b> Audit the configuration of third-party cloud services.	 <b>Badlock Detection</b> Remote and local checks for CVE-2016-2118 and CVE-2016-0128.	 <b>Bash Shellshock Detection</b> Remote and local checks for CVE-2014-6271 and CVE-2014-7169.	 <b>Basic Network Scan</b> A full system scan suitable for any host.	 <b>Credentialed Patch Audit</b> Authenticate to hosts and enumerate missing updates.
 <b>DROWN Detection</b> Remote checks for CVE-2016-0800.	 <b>Host Discovery</b> A simple scan to discover live hosts and open ports.	 <b>Intel AMT Security Bypass</b> Remote and local checks for CVE-2017-5689.	 <b>Internal PCI Network Scan</b> Perform an internal PCI DSS (11.2.1) vulnerability scan.	 <b>Malware Scan</b> Scan for malware on Windows and Unix systems.	 <b>MDM Config Audit</b> Audit the configuration of mobile device managers.
 <b>Mobile Device Scan</b> Assess mobile devices via Microsoft Exchange or an MDM.	 <b>Offline Config Audit</b> Audit the configuration of network devices.	 <b>PCI Quarterly External Scan</b> Approved for quarterly external scanning as required by PCI.	 <b>Policy Compliance Auditing</b> Audit system configurations against a known baseline.	 <b>SCAP and OVAL Auditing</b> Audit systems using SCAP and OVAL definitions.	 <b>Shadow Brokers Scan</b> Scan for vulnerabilities disclosed in the Shadow Brokers leak.
 <b>Spectre and Meltdown</b> Remote and local checks for CVE-2017-5753, CVE-2017-5715, and CVE-2017-5754.	 <b>WannaCry Ransomware</b> Remote and local checks for MS17-010.	 <b>Web Application Tests</b> Scan for published and unknown web vulnerabilities.			

# Nessus – Configuring a Scan

Network Scan / Configuration  
[Back to Scan Report](#)

**Settings**   [Credentials](#)   [Plugins](#)

**BASIC**

[General](#)   [Schedule](#)   [Notifications](#)

**DISCOVERY**

**ASSESSMENT**

**REPORT**

**ADVANCED**

Name: Network Scan

Description: A basic scan of my network

Folder: My Scans

Targets: 192.168.229.0/24

Upload Targets   Add File

[Save](#)   [Cancel](#)

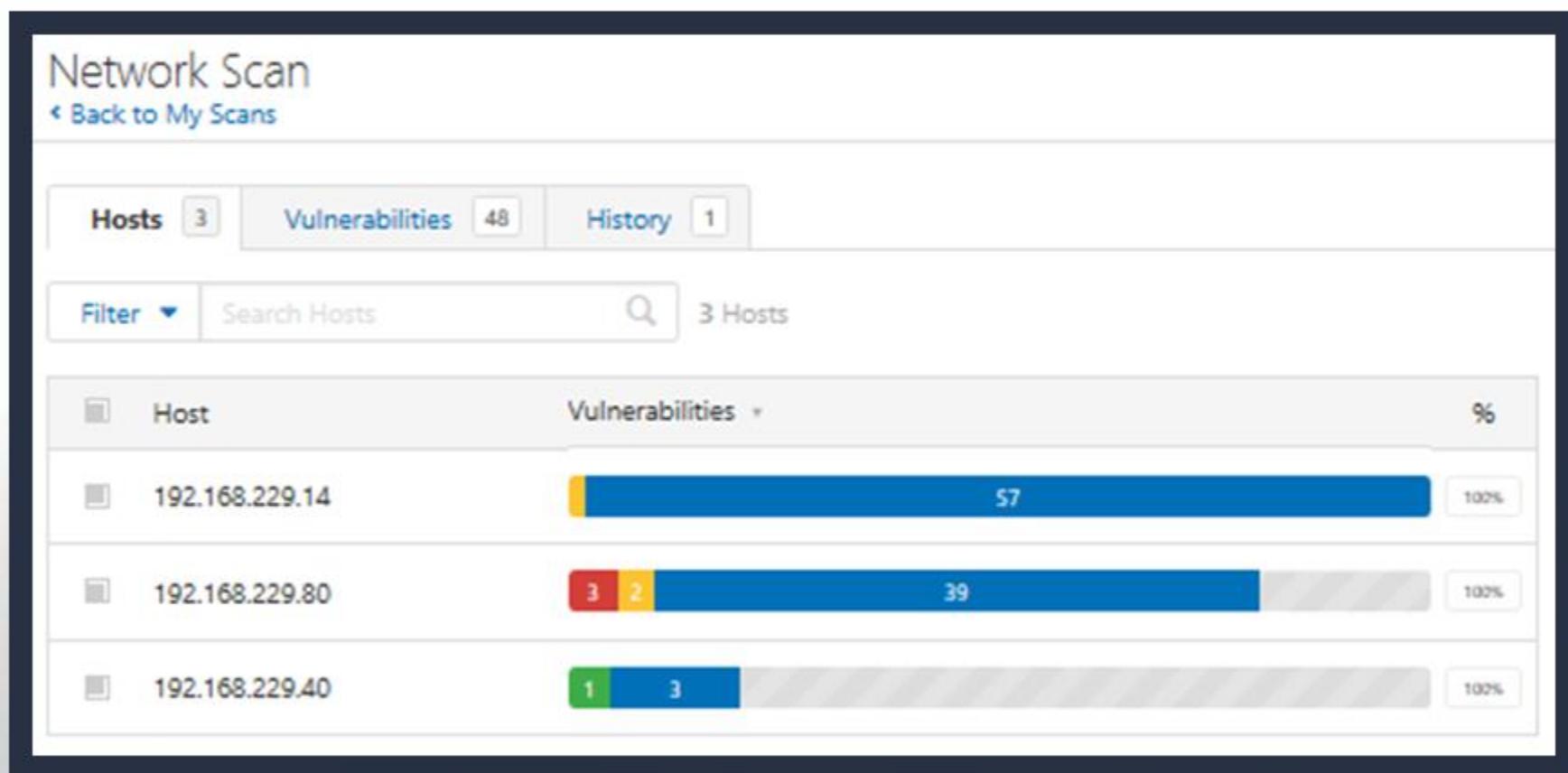
# Nessus – Launching a Scan

The screenshot shows the 'My Scans' page of the Nessus application. At the top, there is a search bar labeled 'Search Scans' with a magnifying glass icon, and a status indicator '3 Scans'. Below the search bar are three scan entries:

Name	Schedule	Last Modified	Actions
192.168.229.80	On Demand	Today at 9:48 AM	▶ X
Basic Scan	On Demand	July 17 at 11:35 AM	<b>Launch</b> ▶ X
Network Scan	On Demand	N/A	▶ X

The 'Basic Scan' entry has a 'Launch' button with a tooltip pointing to it.

# Nessus – Reviewing Scan Results



# Nessus – Reviewing Scan Results

Network Scan / 192.168.229.80 Configure

[Back to Hosts](#)

Vulnerabilities 33 Switch Host 192.168.229.80 ▾

Filter ▾ Search Vulnerabilities Filter 33 Vulnerabilities

Sev	Name	Family	Count
Critical	MS11-030: Vulnerability in DNS Resolution Could...	Windows	1
Critical	MS15-034: Vulnerability in HTTP.sys Could Allow...	Windows	1
Critical	MS17-010: Security Update for Microsoft Window...	Windows	1
Medium	MS16-047: Security Update for SAM and LSAD Re...	Windows	1
Medium	SMB Signing not required	Misc.	1
Info	DCE Services Enumeration	Windows	8

**Host Details**

IP:	192.168.229.80
DNS:	WIN7WS080
MAC:	00:0c:29:e8:6f:4e
OS:	Microsoft Windows 7 Professional
Start:	Today at 10:32 AM
End:	Today at 10:35 AM
Elapsed:	3 minutes

**Vulnerabilities**



- Critical
- High
- Medium
- Low

# Nessus – Reviewing Scan Results

Network Scan

[Back to My Scans](#)

Hosts 3   Vulnerabilities 48   History 1

Filter ▾   Search Hosts  3 Hosts

<input type="checkbox"/> Host	Vulnerabilities
<input type="checkbox"/> 192.168.229.14	<div style="width: 57%;">57</div>
<input type="checkbox"/> 192.168.229.80	<div style="width: 39%;">3 39</div>
<input type="checkbox"/> 192.168.229.40	<div style="width: 3%;">1 3</div>

Configure   Audit Trail   Launch ▾   Export ▾

Nessus   HTML   CSV   Nessus DB

**Scan Details**

Name:	Network Scan
Status:	Completed
Policy:	Basic Network Scan
Scanner:	Local Scanner
Start:	Today at 10:26 AM
End:	Today at 10:45 AM
Elapsed:	19 minutes

## Exercise: Discovery Scanning & Enumerating Hosts

### Objectives

After completing this exercise, students will be able to:

- Conduct active reconnaissance
- Develop mission reports from results of exploitation

### Duration

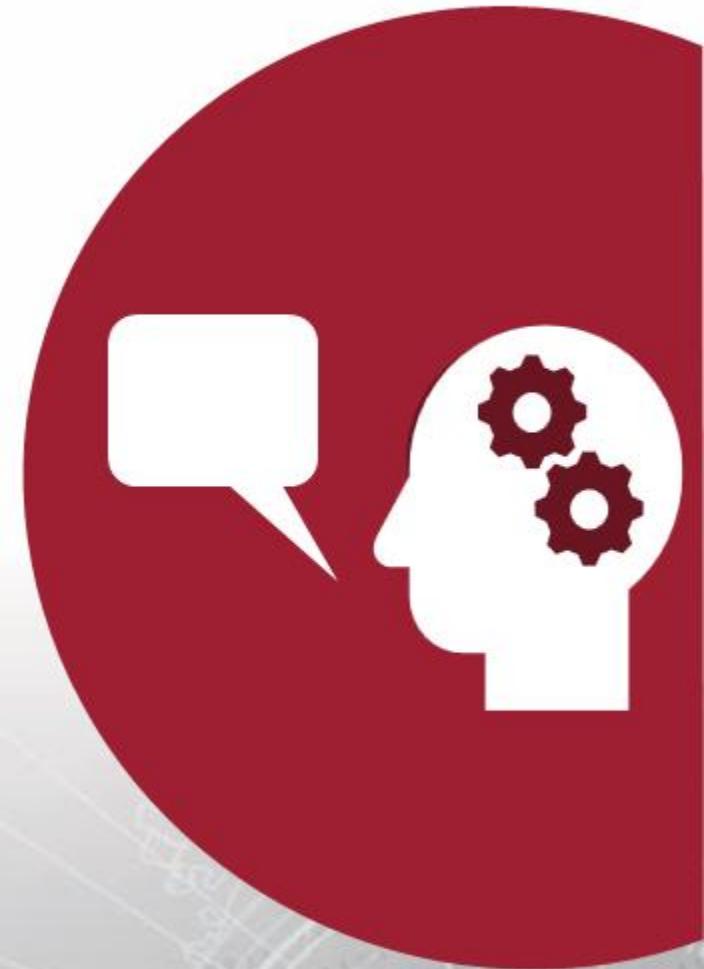
This exercise will take approximately **2** hours to complete.



## Debrief

### Specific Questions

- What are some other ways in which Scapy or a similar tool could be used?



## Debrief

### General Questions

- How did you feel about this procedure?
- Were there any areas in particular where you had difficulty?
- Do you understand how this relates to the work you will be doing?



## Exercise: Advanced Scanning & Evasion Techniques

### Objectives

After completing this exercise, students will be able to:

- Conduct active reconnaissance
- Develop mission reports from results of exploitation

### Duration

This exercise will take approximately **2.5** hours to complete.



## Debrief

### General Questions

- How did you feel about this procedure?
- Were there any areas in particular where you had difficulty?
- Do you understand how this relates to the work you will be doing?

### Specific Questions

- Which other options did you use along with your scans and why did you choose to use them?



## Exercise: Vulnerability Scanning

### Objectives

After completing this exercise, students will be able to:

- Conduct active reconnaissance
- Develop mission reports from results of exploitation

### Duration

This exercise will take approximately **1** hour to complete.



## Debrief

### General Questions

- How did you feel about this procedure?
- Were there any areas in particular where you had difficulty?
- Do you understand how this relates to the work you will be doing?

### Specific Questions

- What is the difference between credentialed and non-credentialed scans?
- How do the Nessus and OpenVAS vulnerability scans compare to the results of previous port scans using Nmap?



## Lesson Summary

In this lesson we learned about:

- Preparing custom packets for scanning
- Using Nmap scans to target ports, detect services, versions of operating systems and applications on a remote host and interpret the results
- Using available scripts to automate networking tasks (such as identifying vulnerabilities, testing controls and detecting backdoors)
- Using Nmap utilities (ncat, ndiff or nping) to analyze a network, generate network packets, connect to other hosts, or compare existing scans
- Performing suitable Nmap scans designed to evade the basic rules of firewalls or Intrusion detection systems
- Completing a vulnerability scan using Nessus and OpenVAS
- Assessing the vulnerability risks to a system and assemble results for inclusion in reporting

End of Module 2, Lesson 2