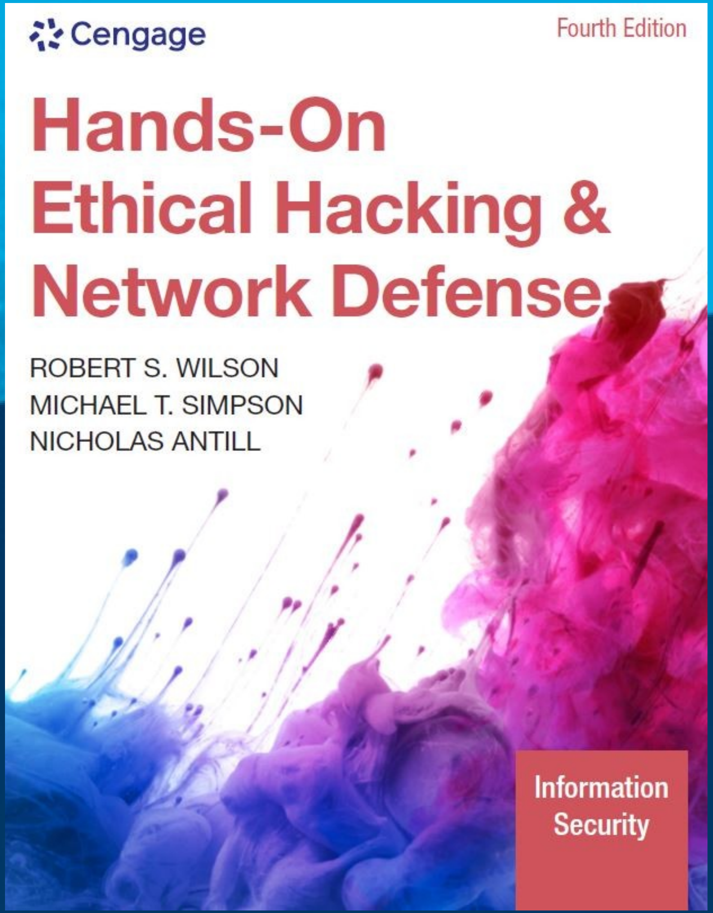


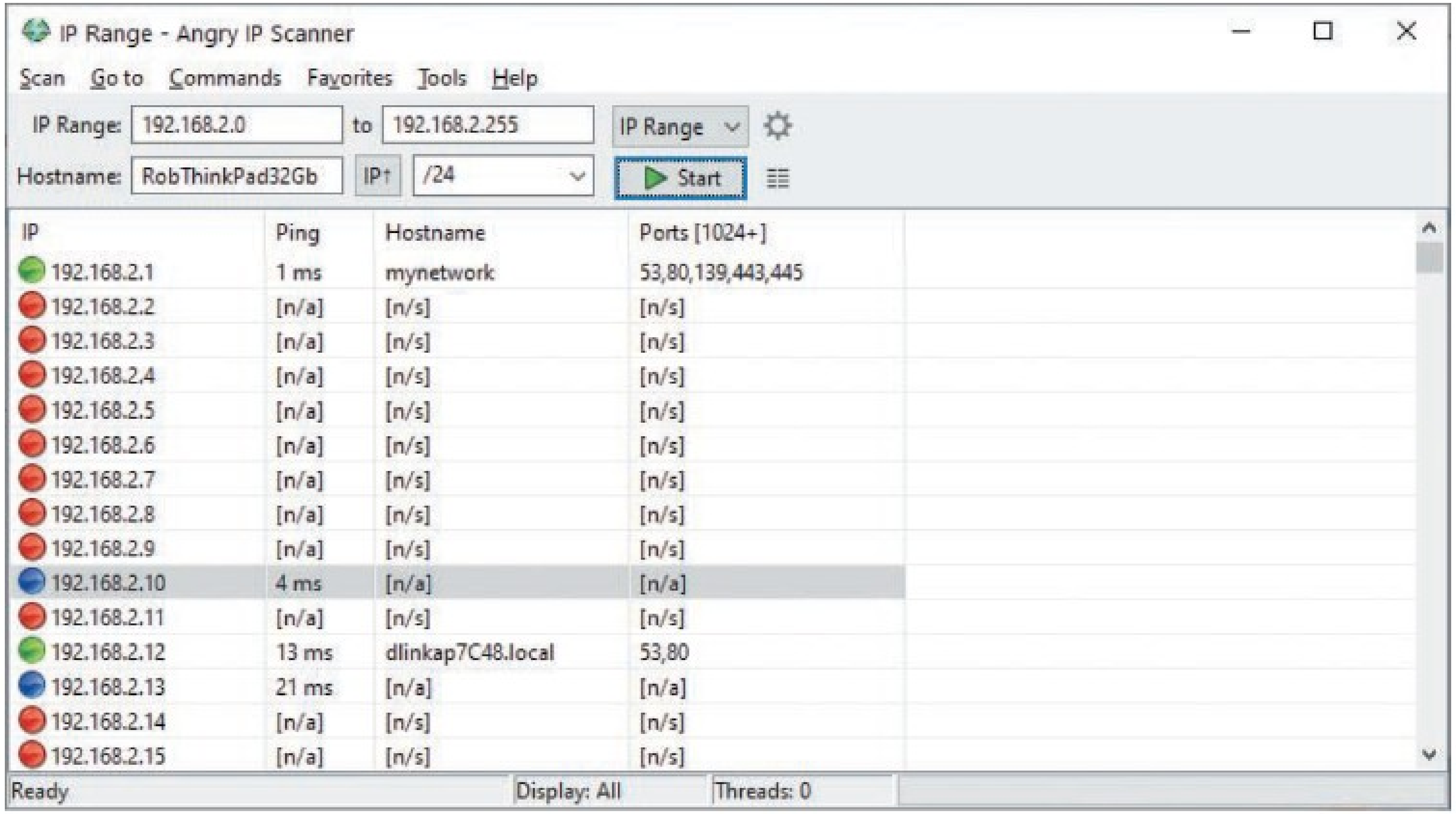
Hands-On Ethical Hacking and Network Defense, Edition 4

Module 5: Port Scanning

Introduction to Port Scanning (1 of 3)

- **Port scanning**
  - Method of finding which services are offered by a host computer
  - Identifies vulnerabilities
- Port-scanning tools
  - Identify vulnerable open ports and launch an exploit to attack the system
- Security testers must scan all ports when testing
  - Not just well-known ports

Introduction to Port Scanning (2 of 3)


IP Range - Angry IP Scanner
Scan Go to Commands Favorites Tools Help
IP Range
IP↑
Start

IP Range: 192.168.2.0 to 192.168.2.255

Hostname: RobThinkPad32Gb /24

IP	Ping	Hostname	Ports [1024+]
🟢 192.168.2.1	1 ms	mynetwork	53,80,139,443,445
🔴 192.168.2.2	[n/a]	[n/s]	[n/s]
🔴 192.168.2.3	[n/a]	[n/s]	[n/s]
🔴 192.168.2.4	[n/a]	[n/s]	[n/s]
🔴 192.168.2.5	[n/a]	[n/s]	[n/s]
🔴 192.168.2.6	[n/a]	[n/s]	[n/s]
🔴 192.168.2.7	[n/a]	[n/s]	[n/s]
🔴 192.168.2.8	[n/a]	[n/s]	[n/s]
🔴 192.168.2.9	[n/a]	[n/s]	[n/s]
🔵 192.168.2.10	4 ms	[n/a]	[n/a]
🔴 192.168.2.11	[n/a]	[n/s]	[n/s]
🟢 192.168.2.12	13 ms	dlinkap7C48.local	53,80
🔵 192.168.2.13	21 ms	[n/a]	[n/a]
🔴 192.168.2.14	[n/a]	[n/s]	[n/s]
🔴 192.168.2.15	[n/a]	[n/s]	[n/s]

Ready Display: All Threads: 0

Source: Angry IP Scanner

Figure 5-1 Angry IP port scanner interface

Introduction to Port Scanning (3 of 3)

- Port-scanning programs report:
  - **Open ports**
    - Allow access to applications and can be vulnerable to an attack
  - **Closed ports**
    - Don’t allow entry or access to a service
  - **Filtered ports**
    - Might indicate that a firewall is being used to allow specified traffic into or out of the network

Types of Port Scans (1 of 2)

- SYN scan
  - Stealthy scan
- Connect scan
  - Completes the three-way handshake
- NULL scan
  - All packet flags are turned off
- XMAS scan
  - FIN, PSH, and URG flags are set

Types of Port Scans (2 of 2)

- ACK scan
  - Used to get past a firewall or other filtering device
- FIN scan
  - Closed port responds with an RST packet when the FIN packet is sent to the target computer
- UDP scan
  - UDP packet is sent to the target computer
    - If port sends back an ICMP “Port Unreachable” message
      - Implies that the port is closed

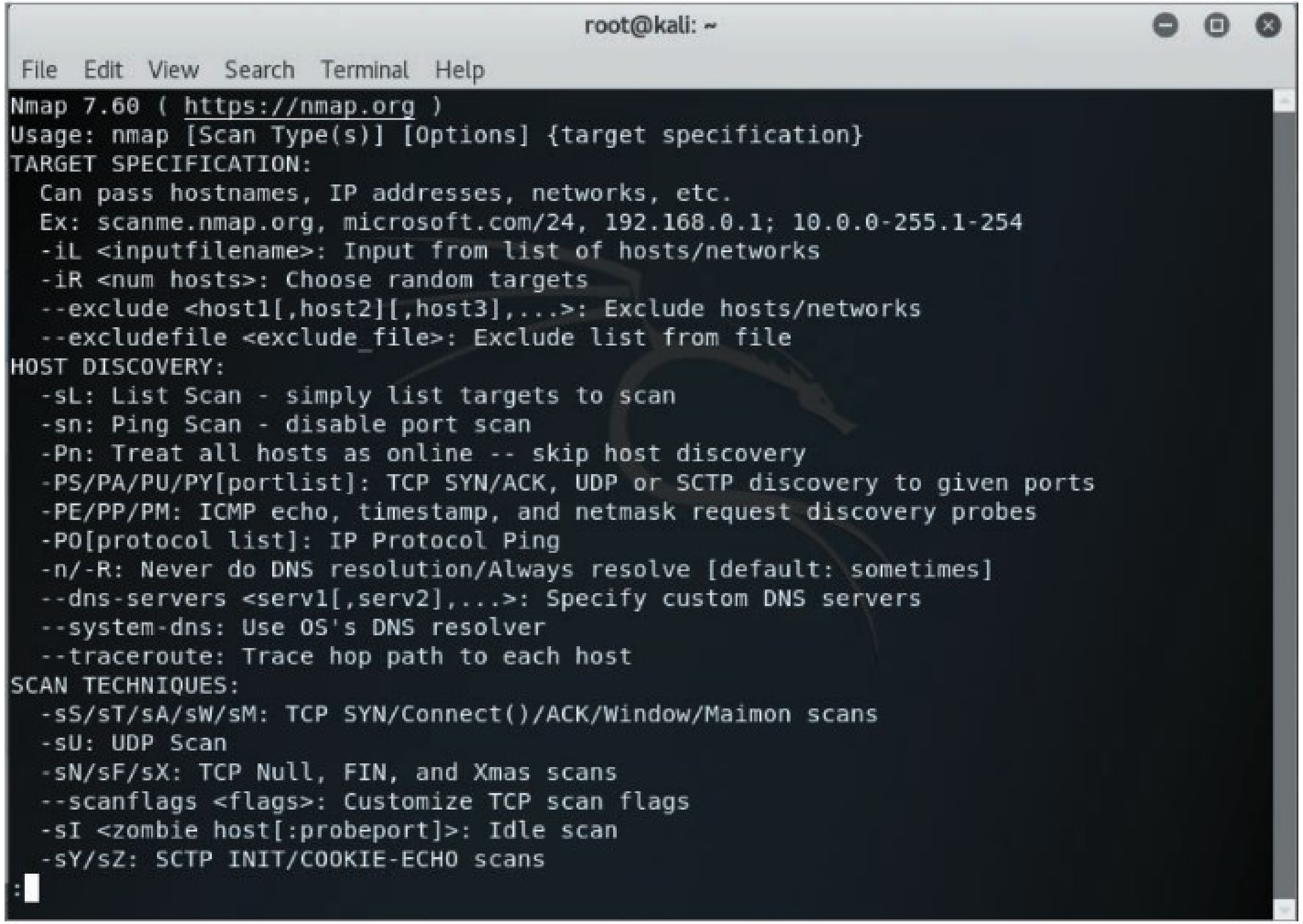
Using Port-Scanning Tools

- Port-scanning tools
  - Hundreds are available
  - Not all are accurate
    - Be familiar with a variety of tools
    - Practice often to gain proficiency
  - Do not use one tool exclusively
- Some tools include:
  - **Nmap**
  - Nessus and OpenVAS

Nmap (1 of 2)

- Originally written for *Phrack* magazine
  - One of the most popular port-scanning tools
  - New features are frequently added
- GUI front end
  - Known as Zenmap
  - Makes working with complex options easier
- Standard port-scanning tool for security professionals
  - Command: `nmap 193.145.85.201`
    - Scans every port on the computer with this IP address

Nmap (2 of 2)



```
Nmap 7.60 ( 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
  -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 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
:
```

Source: Kali Linux

Figure 5-2 Nmap help screen

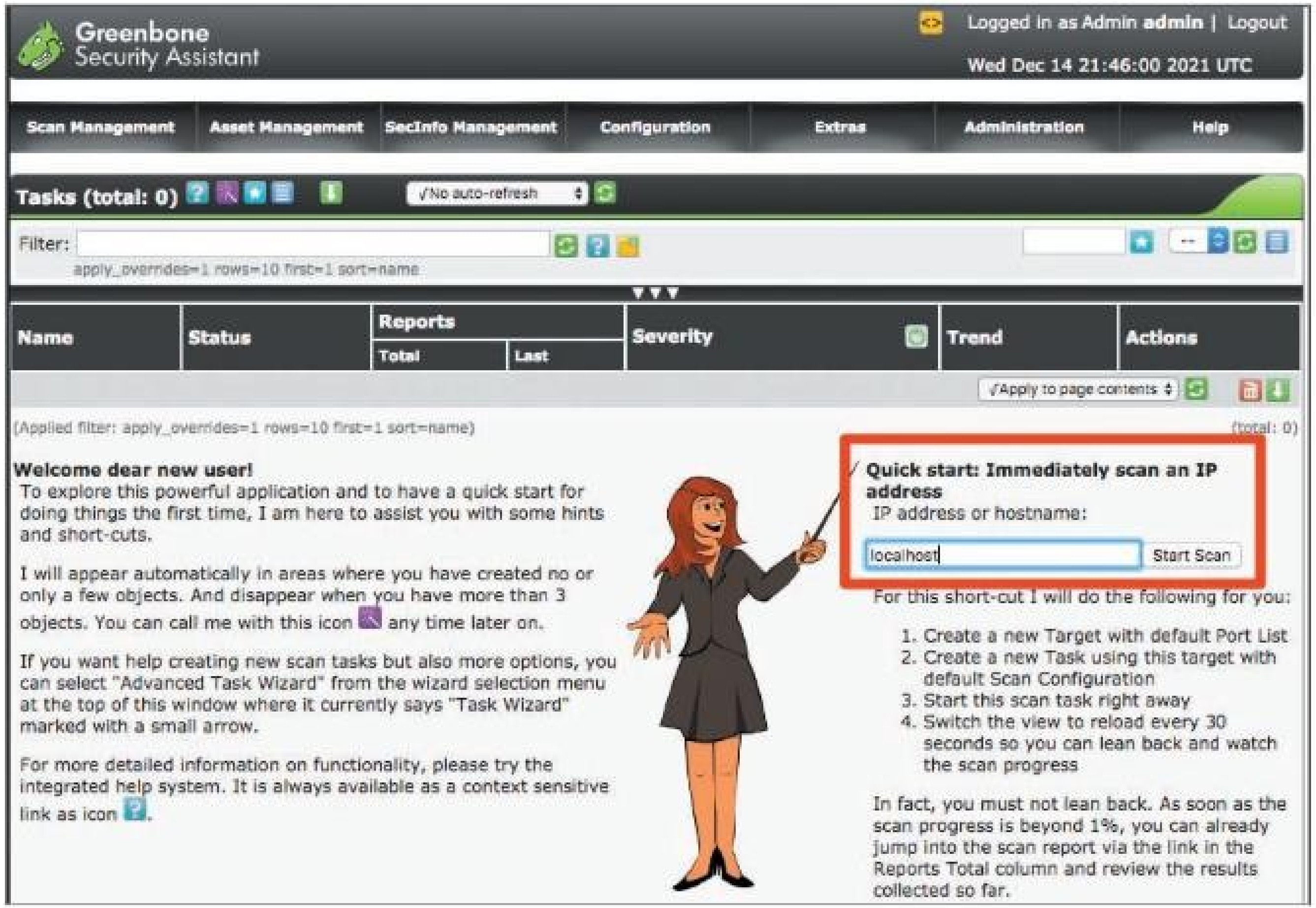
Nessus and OpenVAS (or Greenbone Security Assistant) (1 of 4)

- **Nessus**
  - Vulnerable assessment tool from Tenable
  - Extends NMAP capabilities by analyzing open ports for specific version information
  - Provides detailed vulnerability information on the corresponding service
  - Nessus Professional
    - Product you purchase
  - Nessus Essentials
    - Provides a free version

Nessus and OpenVAS (or Greenbone Security Assistant) (2 of 4)

- **OpenVAS**
  - Open-source fork of Nessus
  - Now branded as Greenbone Security Assistant
  - Capable of updating security check plug-ins when they become available
    - Security test program that can be selected from the client interface
    - Leaving the Safe checks enabled in the policy is advisable
    - Can also determine what vulnerabilities are associated with services

Nessus and OpenVAS (or Greenbone Security Assistant) (3 of 4)



Logged in as Admin **admin** | Logout

Wed Dec 14 21:46:00 2021 UTC

Tasks (total: 0)

apply_overrides=1 rows=10 first=1 sort=name

Name	Status	Reports		Severity	Trend	Actions
		Total	Last			

(Applied filter: apply_overrides=1 rows=10 first=1 sort=name) (total: 0)

Welcome dear new user!

To explore this powerful application and to have a quick start for doing things the first time, I am here to assist you with some hints and short-cuts.

I will appear automatically in areas where you have created no or only a few objects. And disappear when you have more than 3 objects. You can call me with this icon any time later on.

If you want help creating new scan tasks but also more options, you can select "Advanced Task Wizard" from the wizard selection menu at the top of this window where it currently says "Task Wizard" marked with a small arrow.

For more detailed information on functionality, please try the integrated help system. It is always available as a context sensitive link as icon.

Quick start: Immediately scan an IP address

IP address or hostname: localhost

For this short-cut I will do the following for you:

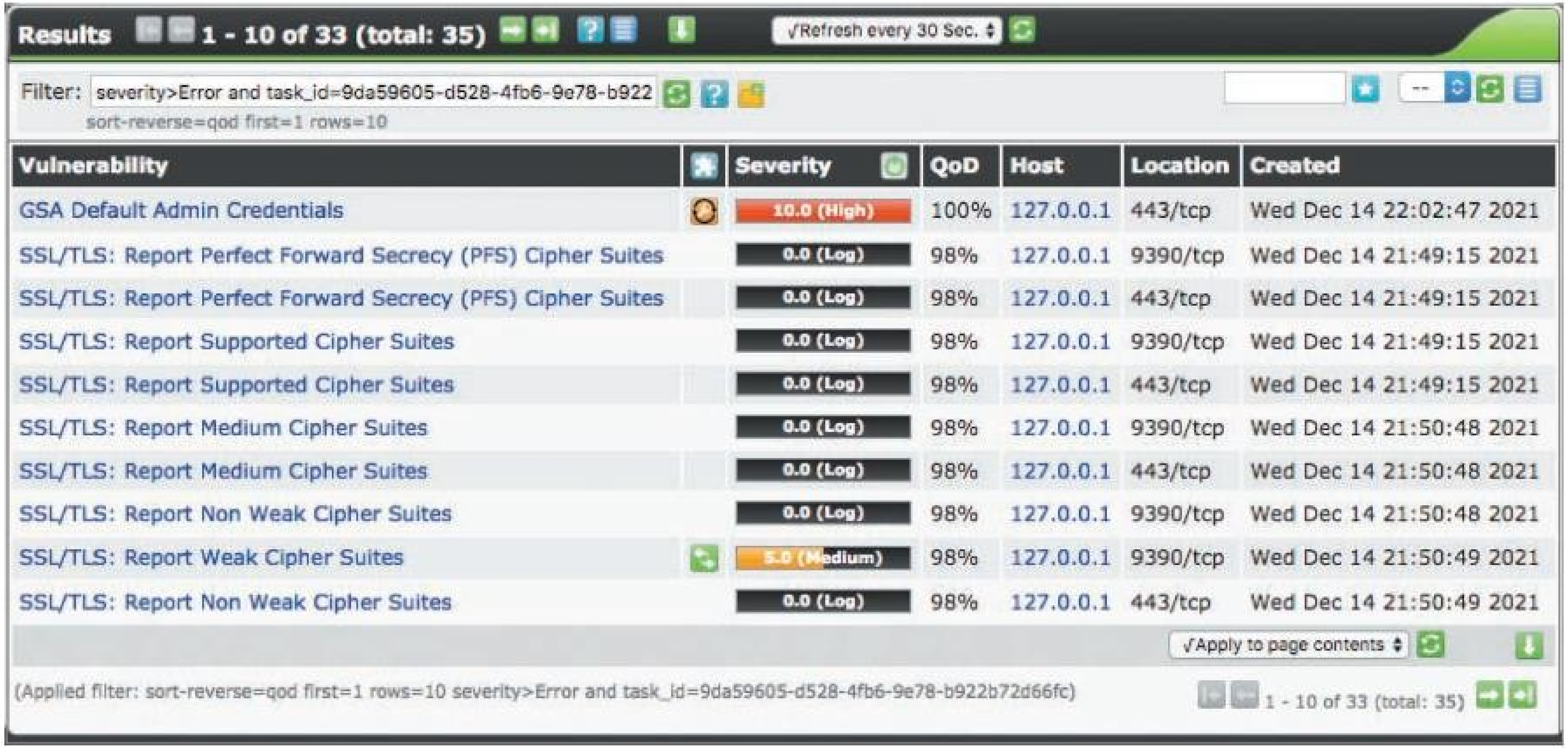
1. Create a new Target with default Port List
2. Create a new Task using this target with default Scan Configuration
3. Start this scan task right away
4. Switch the view to reload every 30 seconds so you can lean back and watch the scan progress

In fact, you must not lean back. As soon as the scan progress is beyond 1%, you can already jump into the scan report via the link in the Reports Total column and review the results collected so far.

Source: GNU General Public License

Figure 5-4 OpenVAS (Greenbone Security Assistant) home screen

Nessus and OpenVAS (or Greenbone Security Assistant) (4 of 4)



Results 1 - 10 of 33 (total: 35)

Filter: severity>Error and task_id=9da59605-d528-4fb6-9e78-b922 sort-reverse=qod first=1 rows=10

Vulnerability		Severity	QoD	Host	Location	Created
GSA Default Admin Credentials		10.0 (High)	100%	127.0.0.1	443/tcp	Wed Dec 14 22:02:47 2021
SSL/TLS: Report Perfect Forward Secrecy (PFS) Cipher Suites		0.0 (Log)	98%	127.0.0.1	9390/tcp	Wed Dec 14 21:49:15 2021
SSL/TLS: Report Perfect Forward Secrecy (PFS) Cipher Suites		0.0 (Log)	98%	127.0.0.1	443/tcp	Wed Dec 14 21:49:15 2021
SSL/TLS: Report Supported Cipher Suites		0.0 (Log)	98%	127.0.0.1	9390/tcp	Wed Dec 14 21:49:15 2021
SSL/TLS: Report Supported Cipher Suites		0.0 (Log)	98%	127.0.0.1	443/tcp	Wed Dec 14 21:49:15 2021
SSL/TLS: Report Medium Cipher Suites		0.0 (Log)	98%	127.0.0.1	9390/tcp	Wed Dec 14 21:50:48 2021
SSL/TLS: Report Medium Cipher Suites		0.0 (Log)	98%	127.0.0.1	443/tcp	Wed Dec 14 21:50:48 2021
SSL/TLS: Report Non Weak Cipher Suites		0.0 (Log)	98%	127.0.0.1	9390/tcp	Wed Dec 14 21:50:48 2021
SSL/TLS: Report Weak Cipher Suites		5.0 (Medium)	98%	127.0.0.1	9390/tcp	Wed Dec 14 21:50:49 2021
SSL/TLS: Report Non Weak Cipher Suites		0.0 (Log)	98%	127.0.0.1	443/tcp	Wed Dec 14 21:50:49 2021

(Applied filter: sort-reverse=qod first=1 rows=10 severity>Error and task_id=9da59605-d528-4fb6-9e78-b922b72d66fc)

1 - 10 of 33 (total: 35)

Figure 5-5 Vulnerabilities listed in OpenVAS

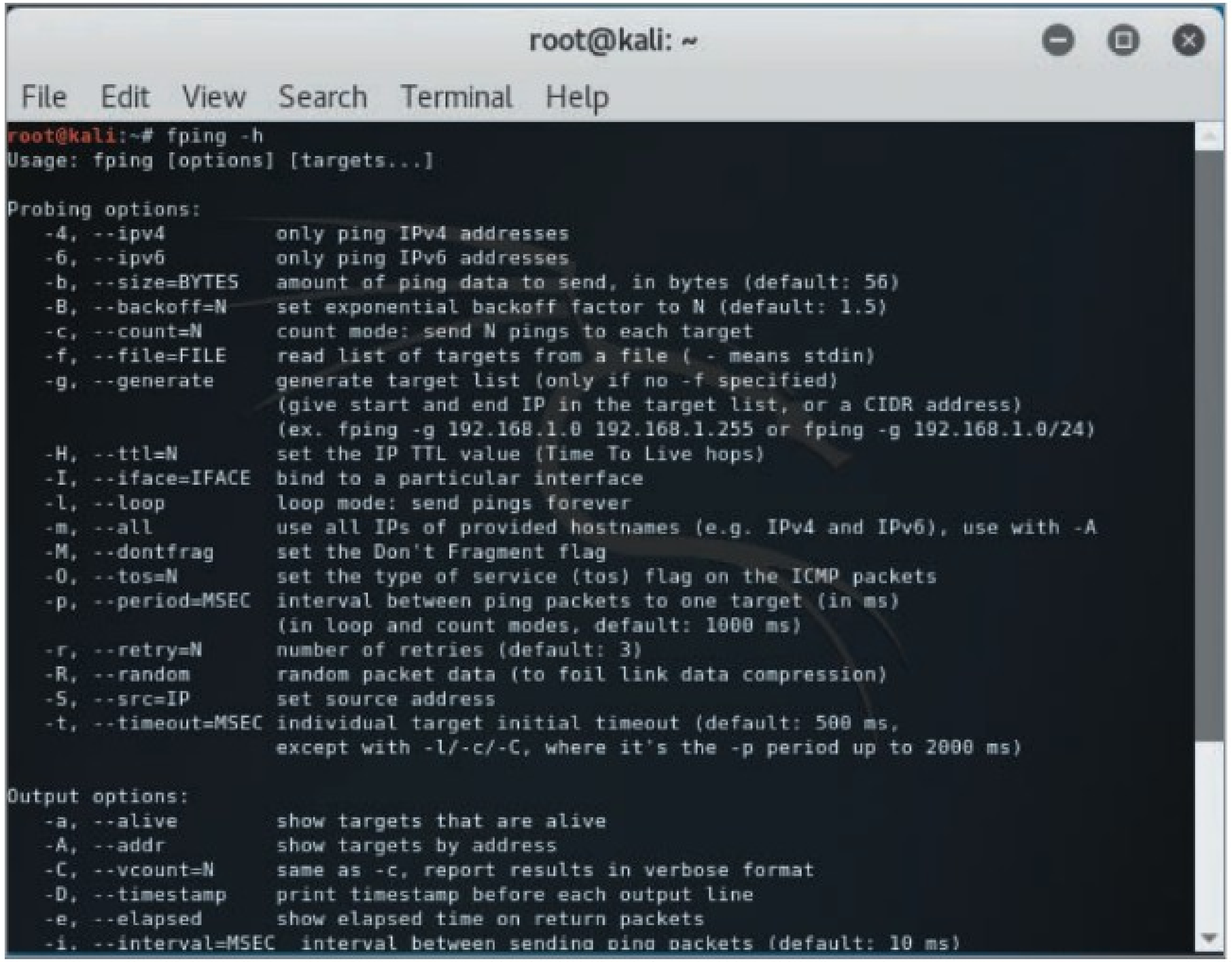
Conducting Ping Sweeps

- **Ping sweeps**
  - Identify which IP addresses belong to active hosts
    - Ping a range of IP addresses to see what type of response is returned
- Problems
  - Might shut down computers at the time of the sweep
    - Indicates that the IP address does not belong to a live host
  - Many network administrators configure nodes to not respond to an ICMP Echo Request (type 8) with an ICMP Echo Reply (type 0)
  - Firewalls may filter out ICMP traffic

Fping (1 of 4)

- With the **Fping** tool, you can ping multiple IP addresses simultaneously
  - Included with Kali Linux
- Accepts a range of IP addresses
  - Entered at a command prompt
  - You can create a file containing multiple IP addresses
    - Use it as input for the `Fping` command
- Input file
  - Usually created with a shell-scripting language so that you don't need to type thousands of IP addresses needed for a ping sweep

Fping (2 of 4)



```
root@kali:~# fping -h
Usage: fping [options] [targets...]

Probing options:
   -4, --ipv4         only ping IPv4 addresses
   -6, --ipv6         only ping IPv6 addresses
   -b, --size=BYTES   amount of ping data to send, in bytes (default: 56)
   -B, --backoff=N    set exponential backoff factor to N (default: 1.5)
   -c, --count=N      count mode: send N pings to each target
   -f, --file=FILE    read list of targets from a file ( - means stdin)
   -g, --generate     generate target list (only if no -f specified)
                      (give start and end IP in the target list, or a CIDR address)
                      (ex. fping -g 192.168.1.0 192.168.1.255 or fping -g 192.168.1.0/24)
   -H, --ttl=N        set the IP TTL value (Time To Live hops)
   -I, --iface=IFACE  bind to a particular interface
   -l, --loop         loop mode: send pings forever
   -m, --all          use all IPs of provided hostnames (e.g. IPv4 and IPv6), use with -A
   -M, --dontfrag     set the Don't Fragment flag
   -O, --tos=N        set the type of service (tos) flag on the ICMP packets
   -p, --period=MSEC  interval between ping packets to one target (in ms)
                      (in loop and count modes, default: 1000 ms)
   -r, --retry=N      number of retries (default: 3)
   -R, --random       random packet data (to foil link data compression)
   -S, --src=IP       set source address
   -t, --timeout=MSEC individual target initial timeout (default: 500 ms,
                      except with -l/-c/-C, where it's the -p period up to 2000 ms)

Output options:
   -a, --alive        show targets that are alive
   -A, --addr         show targets by address
   -C, --vcount=N     same as -c, report results in verbose format
   -D, --timestamp    print timestamp before each output line
   -e, --elapsed      show elapsed time on return packets
   -i, --interval=MSEC  interval between sending ping packets (default: 10 ms)
```

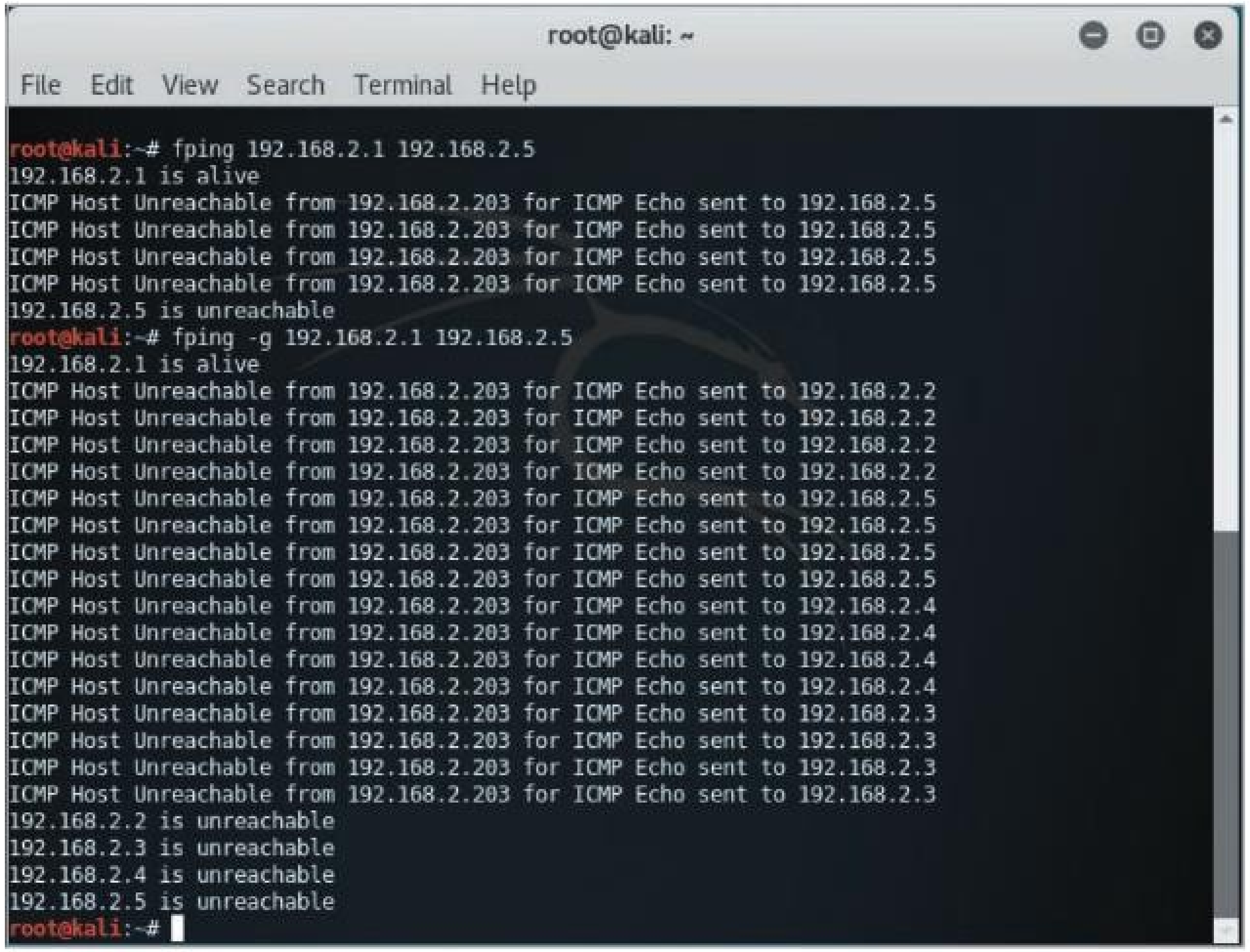
Source: GNU Public License

Figure 5-6 Fping parameters

Fping (3 of 4)

- To ping sweep a range of IP addresses without using an input file, use the command:
  - `fping –g BeginningIPaddress EndingIPaddress`
  - The `–g` parameter is used when no input file is available
  - Example:
    - `fping –g 192.168.185.1 192.168.185.5` command returns the results shown on Figure 5-6

Fping (4 of 4)



```
root@kali:~# fping 192.168.2.1 192.168.2.5
192.168.2.1 is alive
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
192.168.2.5 is unreachable
root@kali:~# fping -g 192.168.2.1 192.168.2.5
192.168.2.1 is alive
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.2
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.2
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.2
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.2
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.5
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.4
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.4
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.4
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.4
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.3
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.3
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.3
ICMP Host Unreachable from 192.168.2.203 for ICMP Echo sent to 192.168.2.3
192.168.2.2 is unreachable
192.168.2.3 is unreachable
192.168.2.4 is unreachable
192.168.2.5 is unreachable
root@kali:~#
```

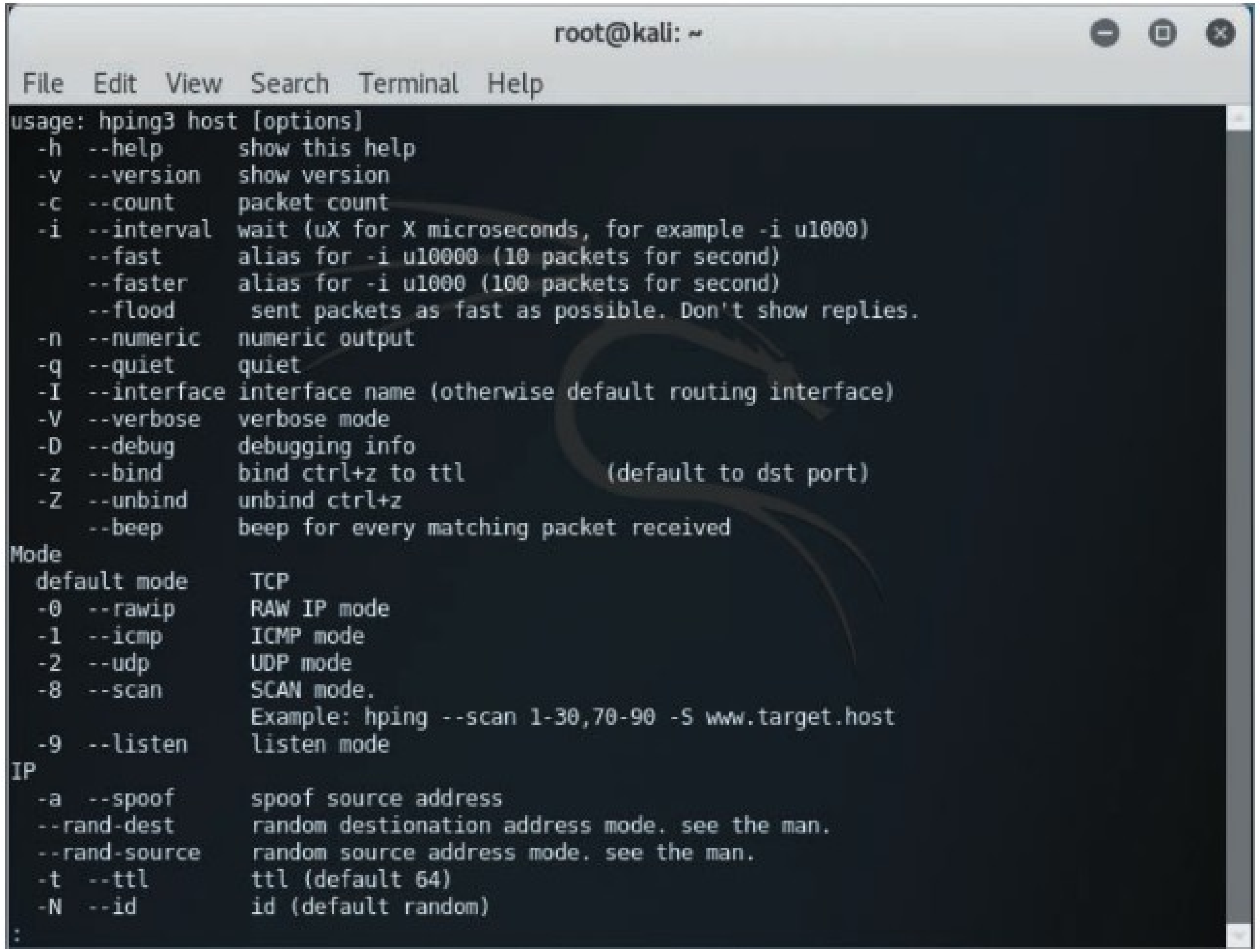
Source: GNU General Public License

Figure 5-7 Results of fping commands

Hping3 (1 of 4)

- Used to:
  - Perform ping sweeps
  - Bypass filtering devices
    - Allows users to inject modified IP packets
- Advanced port-scanning tool
  - All security testers must be familiar with this tool
  - Offers a variety of features

Hping3 (2 of 4)

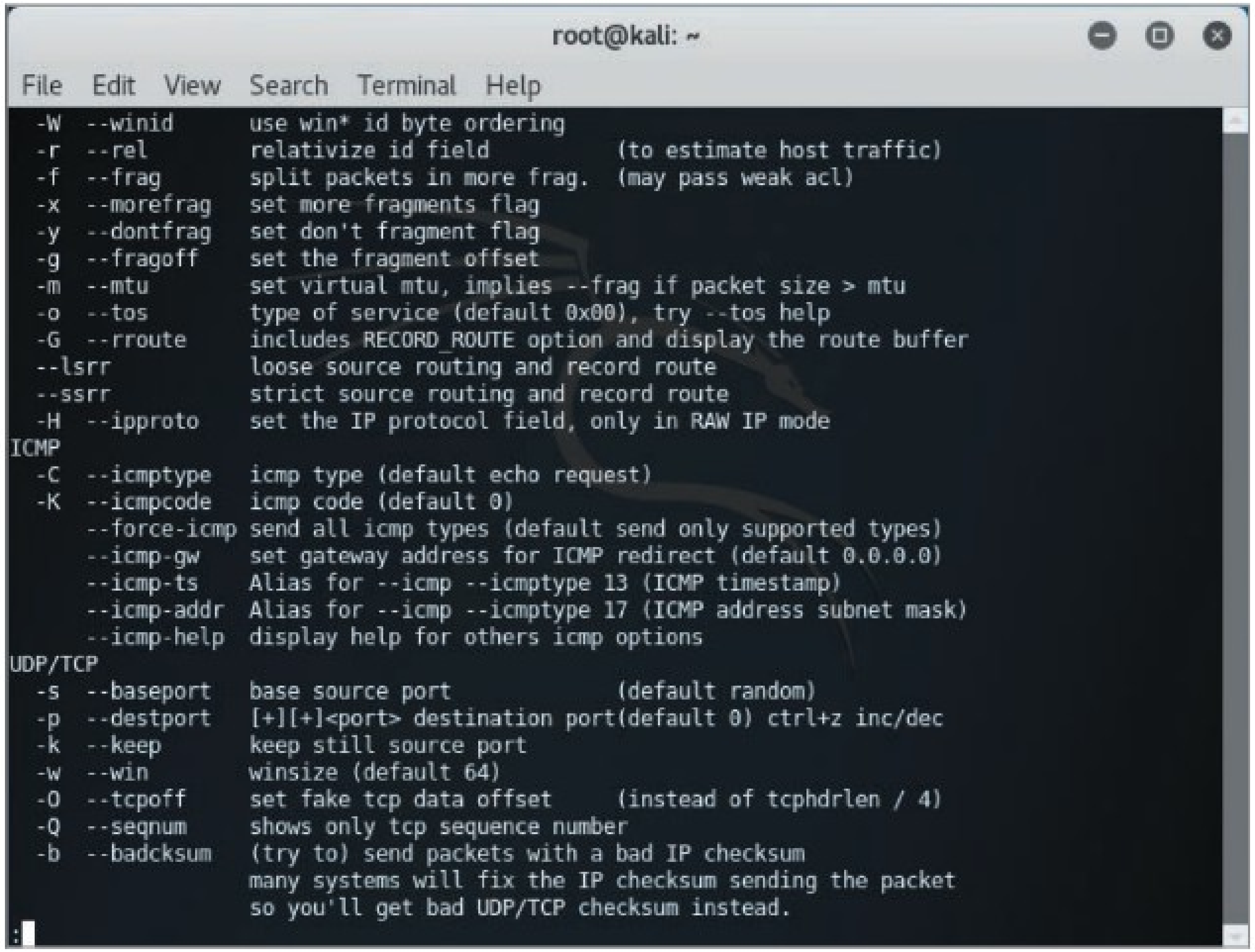


```
usage: hping3 host [options]
  -h  --help      show this help
  -v  --version   show version
  -c  --count     packet count
  -i  --interval  wait (uX for X microseconds, for example -i u1000)
      --fast      alias for -i u10000 (10 packets for second)
      --faster    alias for -i u1000 (100 packets for second)
      --flood      sent packets as fast as possible. Don't show replies.
  -n  --numeric   numeric output
  -q  --quiet     quiet
  -I  --interface interface name (otherwise default routing interface)
  -V  --verbose   verbose mode
  -D  --debug     debugging info
  -z  --bind      bind ctrl+z to ttl           (default to dst port)
  -Z  --unbind    unbind ctrl+z
      --beep      beep for every matching packet received
Mode
  default mode     TCP
  -0  --rawip      RAW IP mode
  -1  --icmp       ICMP mode
  -2  --udp        UDP mode
  -8  --scan       SCAN mode.
                   Example: hping --scan 1-30,70-90 -S www.target.host
  -9  --listen     listen mode
IP
  -a  --spoof      spoof source address
  --rand-dest      random destionation address mode. see the man.
  --rand-source    random source address mode. see the man.
  -t  --ttl        ttl (default 64)
  -N  --id         id (default random)
:
```

Source: GNU General Public License

Figure 5-8 Hping3 help page 1

Hping3 (3 of 4)

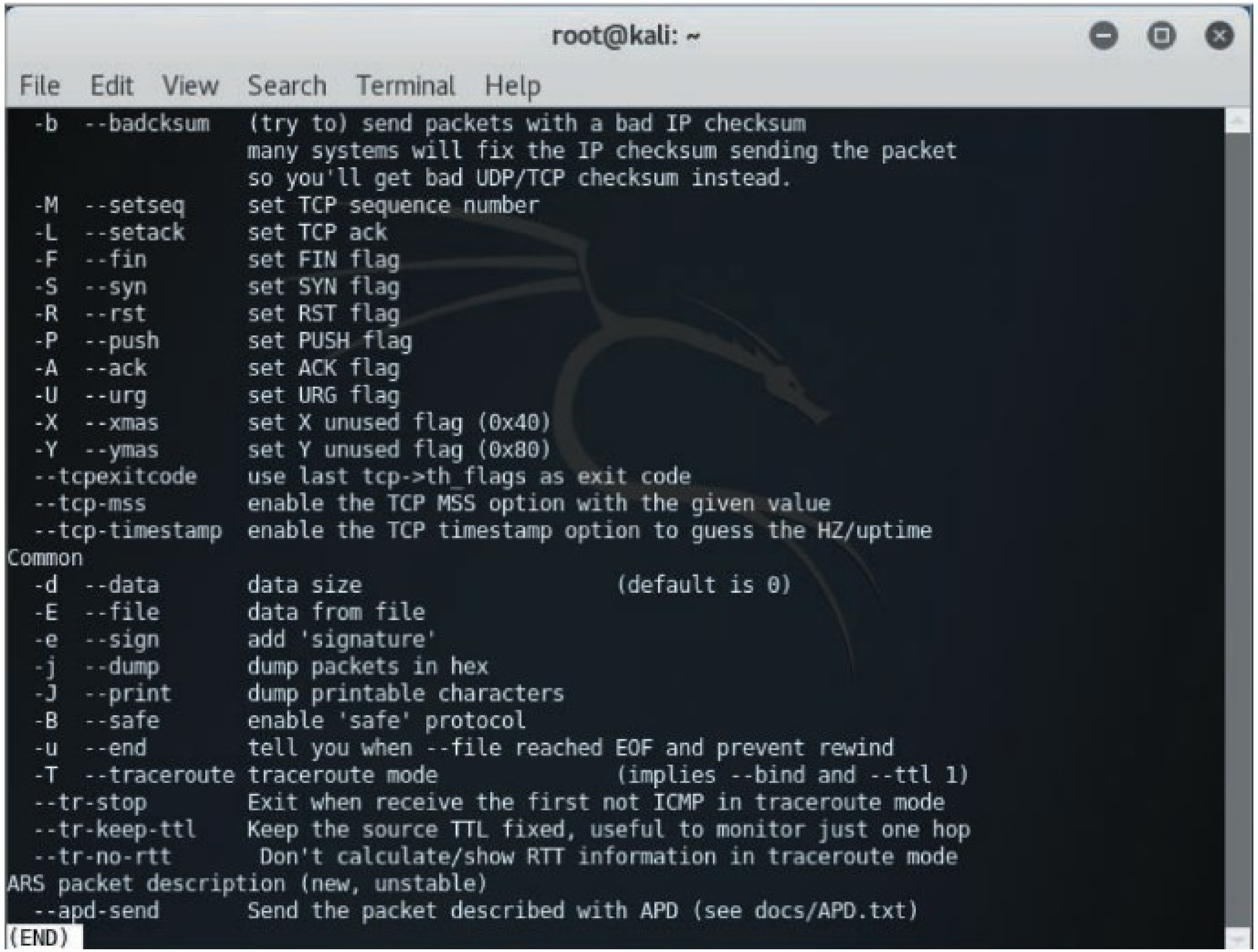


```
  -W  --winid       use win* id byte ordering
  -r  --rel         relativize id field          (to estimate host traffic)
  -f  --frag        split packets in more frag.  (may pass weak acl)
  -x  --morefrag    set more fragments flag
  -y  --dontfrag    set don't fragment flag
  -g  --fragoff     set the fragment offset
  -m  --mtu         set virtual mtu, implies --frag if packet size > mtu
  -o  --tos         type of service (default 0x00), try --tos help
  -G  --rroute      includes RECORD_ROUTE option and display the route buffer
  --lsrr            loose source routing and record route
  --ssrr            strict source routing and record route
  -H  --ipproto     set the IP protocol field, only in RAW IP mode
ICMP
  -C  --icmptype    icmp type (default echo request)
  -K  --icmpcode    icmp code (default 0)
      --force-icmp  send all icmp types (default send only supported types)
      --icmp-gw     set gateway address for ICMP redirect (default 0.0.0.0)
      --icmp-ts     Alias for --icmp --icmptype 13 (ICMP timestamp)
      --icmp-addr   Alias for --icmp --icmptype 17 (ICMP address subnet mask)
      --icmp-help   display help for others icmp options
UDP/TCP
  -s  --baseport    base source port             (default random)
  -p  --destport    [+][+]<port> destination port(default 0) ctrl+z inc/dec
  -k  --keep        keep still source port
  -w  --win         winsize (default 64)
  -O  --tcpoff      set fake tcp data offset     (instead of tcphdrlen / 4)
  -Q  --seqnum      shows only tcp sequence number
  -b  --badcksum    (try to) send packets with a bad IP checksum
                    many systems will fix the IP checksum sending the packet
                    so you'll get bad UDP/TCP checksum instead.
:
```

Source: GNU General Public License

Figure 5-9 Hping3 help page 2

Hping3 (4 of 4)



```
  -b  --badcksum   (try to) send packets with a bad IP checksum
                   many systems will fix the IP checksum sending the packet
                   so you'll get bad UDP/TCP checksum instead.
  -M  --setseq     set TCP sequence number
  -L  --setack     set TCP ack
  -F  --fin        set FIN flag
  -S  --syn        set SYN flag
  -R  --rst        set RST flag
  -P  --push       set PUSH flag
  -A  --ack        set ACK flag
  -U  --urg        set URG flag
  -X  --xmas       set X unused flag (0x40)
  -Y  --ymas       set Y unused flag (0x80)
  --tcpexitcode    use last tcp->th_flags as exit code
  --tcp-mss        enable the TCP MSS option with the given value
  --tcp-timestamp  enable the TCP timestamp option to guess the HZ/uptime
Common
  -d  --data       data size                     (default is 0)
  -E  --file       data from file
  -e  --sign       add 'signature'
  -j  --dump       dump packets in hex
  -J  --print      dump printable characters
  -B  --safe       enable 'safe' protocol
  -u  --end        tell you when --file reached EOF and prevent rewind
  -T  --traceroute traceroute mode               (implies --bind and --ttl 1)
  --tr-stop        Exit when receive the first not ICMP in traceroute mode
  --tr-keep-ttl    Keep the source TTL fixed, useful to monitor just one hop
  --tr-no-rtt       Don't calculate/show RTT information in traceroute mode
ARS packet description (new, unstable)
  --apd-send       Send the packet described with APD (see docs/APD.txt)
(END)
```

Source: GNU General Public License

Figure 5-10 Hping3 help page 3

Crafting IP Packets

- Packets contain:
  - Source IP addresses
  - Destination IP addresses
  - Information about flags
- Helpful tools for crafting IP packets
  - Hping3
  - Fping

Understanding Scripting

- Some tools might need to be modified to better suit your needs as a security tester
- Customized scripts
  - Automates tasks
  - Time-saving
  - Requires basic programming skills

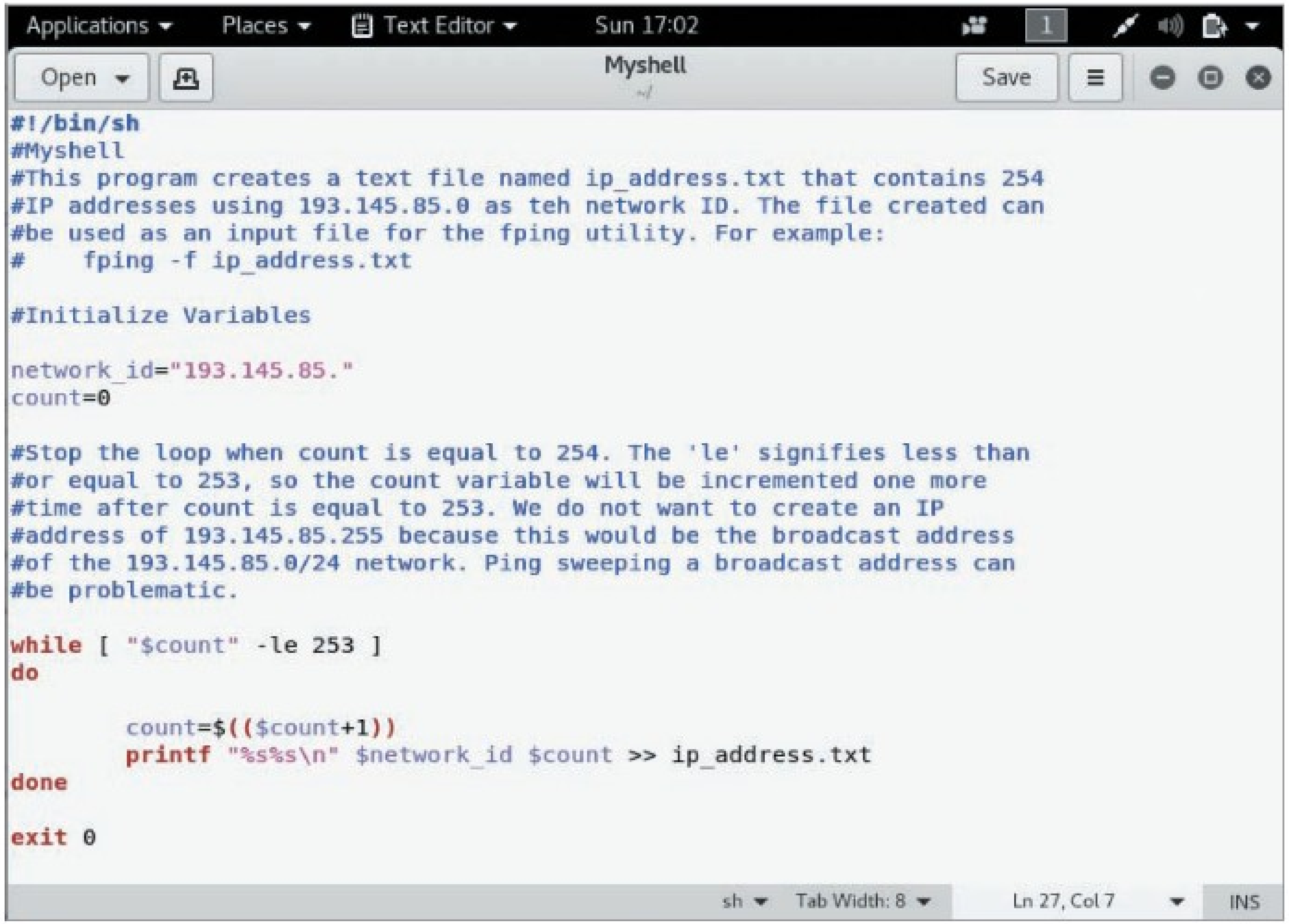
Scripting Basics (1 of 2)

- Similar to DOS batch programming
- A script or batch file
  - Text file that contains multiple commands that are usually entered manually at the command prompt
- If you find that you are using repetitive commands to perform the same task, that task is a good candidate for scripting
- Best way to learn how to create a script
  - Create a script by doing it

Summary of Vim Commands

vim command	Description
A	Appends text after the insertion point
I	Inserts text before the insertion point
Delete key	Overwrites the last character when in Insert mode
X	Deletes the current character
Dd	Deletes the current line
Dw	Deletes the current word
P	Replaces the previously deleted text
Wq	Writes changes and quits the edit session
ZZ	Exits vi and saves all changes

Scripting Basics (2 of 2)



```
#!/bin/sh
#Myshell
#This program creates a text file named ip_address.txt that contains 254
#IP addresses using 193.145.85.0 as teh network ID. The file created can
#be used as an input file for the fping utility. For example:
#    fping -f ip_address.txt

#Initialize Variables

network_id="193.145.85."
count=0

#Stop the loop when count is equal to 254. The 'le' signifies less than
#or equal to 253, so the count variable will be incremented one more
#time after count is equal to 253. We do not want to create an IP
#address of 193.145.85.255 because this would be the broadcast address
#of the 193.145.85.0/24 network. Ping sweeping a broadcast address can
#be problematic.

while [ "$count" -le 253 ]
do

        count=$(($count+1))
        printf "%s%s\n" $network_id $count >> ip_address.txt
done

exit 0
```

Source: Kali Linux gedit

Figure 5-11 Shell script with comments

Summary

- Now that the lesson has ended, you should be able to:
  - Describe port scanning and types of port scans
  - Describe port-scanning tools
  - Explain what ping sweeps are used for
  - Explain how shell scripting is used to automate security tasks