

Experiment 3

Aim

To study network analysis and monitoring tools such as: Wire shark, Nmap, Hping.

Theory

Wireshark

Wireshark is the world's foremost and widely-used network protocol analyzer. It lets you see what's happening on your network at a microscopic level and is the de facto (and often de jure) standard across many commercial and non-profit enterprises, government agencies, and educational institutions. Wireshark development thrives thanks to the volunteer contributions of networking experts around the globe and is the continuation of a project started by Gerald Combs in 1998.

Wireshark Features

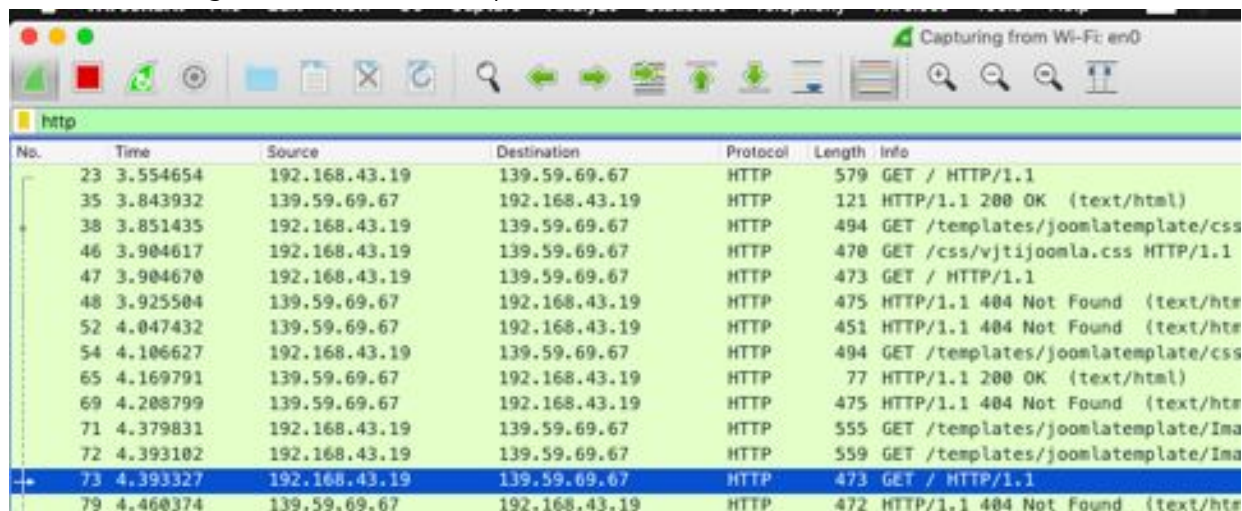
- Deep inspection of hundreds of protocols, with more being added all the time
- Live capture and offline analysis
- Standard three-pane packet browser
- Multi-platform: Runs on Windows, Linux, macOS, Solaris, FreeBSD, NetBSD, and many others
- Captured network data can be browsed via a GUI, or via the TTY-mode TShark utility
- The most powerful display filters in the industry
- Rich VoIP analysis
- Read/write many different capture file formats: tcpdump (libpcap), Pcap NG, Catapult DCT2000, Cisco Secure IDS iplog, Microsoft Network Monitor, Network General Sniffer® (compressed and uncompressed), Sniffer® Pro, and NetXray®, Network Instruments Observer, NetScreen snoop, Novell LANalyzer, RADCOM WAN/LAN Analyzer, Shomiti/Finisar Surveyor, Tektronix K12xx, Visual Networks Visual UpTime, WildPackets EtherPeek/TokenPeek/AiroPeek, and many others
- Capture files compressed with gzip can be decompressed on the fly
- Live data can be read from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, FDDI, and others (depending on your platform)
- Decryption support for many protocols, including IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP, and WPA/WPA2
- Coloring rules can be applied to the packet list for quick, intuitive analysis
- Output can be exported to XML, PostScript®, CSV, or plain text

TEST SERVER	<ol style="list-style-type: none"> 1. http://viti.ac.in 2. https://en.wikipedia.org/ 3. http://norvig.com/big.txt
BROWSER/ HTTP SOFTWARE	<ol style="list-style-type: none"> 1. Chrome Version 75.0.3770.142 (Official Build) (64-bit) 2. Postman Version 7.3.4 (7.3.4)
TEST PLATFORM	macOS Mojave 10.14.6

Q1. The basic HTTP GET response interaction:

- Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?

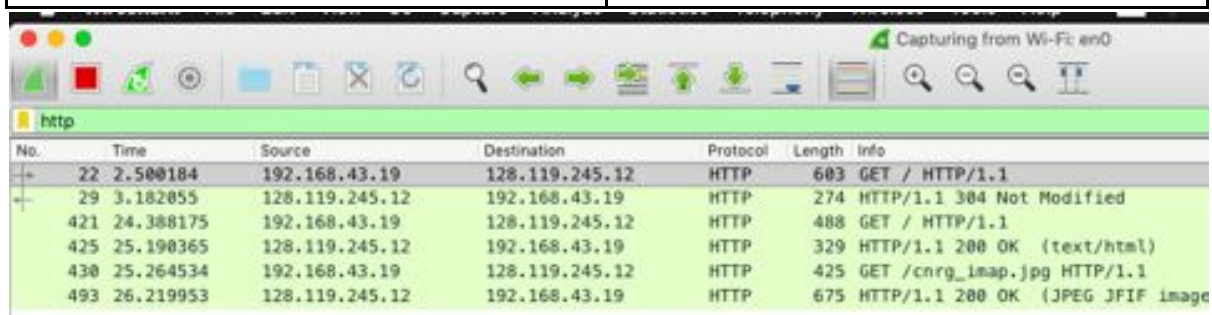
Browser is using HTTP 1.1 ; Server responds with HTTP 1.1



No.	Time	Source	Destination	Protocol	Length	Info
23	3.554654	192.168.43.19	139.59.69.67	HTTP	579	GET / HTTP/1.1
35	3.843932	139.59.69.67	192.168.43.19	HTTP	121	HTTP/1.1 200 OK (text/html)
38	3.851435	192.168.43.19	139.59.69.67	HTTP	494	GET /templates/joomltemplate/css
46	3.904617	192.168.43.19	139.59.69.67	HTTP	470	GET /css/vjti Joomla.css HTTP/1.1
47	3.904670	192.168.43.19	139.59.69.67	HTTP	473	GET / HTTP/1.1
48	3.925504	139.59.69.67	192.168.43.19	HTTP	475	HTTP/1.1 404 Not Found (text/html)
52	4.047432	139.59.69.67	192.168.43.19	HTTP	451	HTTP/1.1 404 Not Found (text/html)
54	4.106627	192.168.43.19	139.59.69.67	HTTP	494	GET /templates/joomltemplate/css
65	4.169791	139.59.69.67	192.168.43.19	HTTP	77	HTTP/1.1 200 OK (text/html)
69	4.208799	139.59.69.67	192.168.43.19	HTTP	475	HTTP/1.1 404 Not Found (text/html)
71	4.379831	192.168.43.19	139.59.69.67	HTTP	555	GET /templates/joomltemplate/Ima
72	4.393102	192.168.43.19	139.59.69.67	HTTP	559	GET /templates/joomltemplate/Ima
73	4.393327	192.168.43.19	139.59.69.67	HTTP	473	GET / HTTP/1.1
79	4.460374	139.59.69.67	192.168.43.19	HTTP	472	HTTP/1.1 404 Not Found (text/html)

- What is the IP address of your computer? Of the gaia.cs.umass.edu server?

CLIENT IP	192.168.43.19
SERVER IP	128.119.245.12



No.	Time	Source	Destination	Protocol	Length	Info
22	2.500184	192.168.43.19	128.119.245.12	HTTP	603	GET / HTTP/1.1
29	3.182055	128.119.245.12	192.168.43.19	HTTP	274	HTTP/1.1 304 Not Modified
421	24.388175	192.168.43.19	128.119.245.12	HTTP	488	GET / HTTP/1.1
425	25.190365	128.119.245.12	192.168.43.19	HTTP	329	HTTP/1.1 200 OK (text/html)
430	25.264534	192.168.43.19	128.119.245.12	HTTP	425	GET /cnrg_imap.jpg HTTP/1.1
493	26.219953	128.119.245.12	192.168.43.19	HTTP	675	HTTP/1.1 200 OK (JPEG JFIF image)

- When was the HTML file that you are retrieving last modified at the server?

“Last-Modified: Tue, 01 Mar 2016 18:57:50 GMT\r\n”

```

> Ethernet II, Src: d2:04:01:c4:d1:20 (d2:04:01:c4:d1:20), Dst: Apple_5e:89:4c (60:30:d4:5e:89:4c)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.43.19
> Transmission Control Protocol, Src Port: 80, Dst Port: 62886, Seq: 2717, Ack: 423, Len: 263
> [3 Reassembled TCP Segments (2979 bytes): #423(1358), #424(1358), #425(263)]
▼ Hypertext Transfer Protocol
  ▶ HTTP/1.1 200 OK\r\n
    Date: Thu, 22 Aug 2019 04:48:46 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16 mod_perl/2.0.10 Perl/v5.16.3\r\n
    Last-Modified: Tue, 01 Mar 2016 18:57:50 GMT\r\n
    ETag: "a5b-52d015789ee9e"\r\n
    Accept-Ranges: bytes\r\n
    Content-Type: text/html; charset=UTF-8\r\n
  ▶ Content-Length: 2651\r\n
    Connection: keep-alive\r\n
    \r\n
    [HTTP response 1/2]
    [Time since request: 0.802190000 seconds]
    [Request in frame: 421]
    [Next request in frame: 430]
    [Next response in frame: 493]
    File Data: 2651 bytes
  ▶ Line-based text data: text/html (68 lines)

```

- How many bytes of content are being returned to your browser?

2651 bytes (refer image above)

Q2. The HTTP CONDITIONAL GET/response interaction:

- Inspect the content of the server response. Did the server explicitly return the contents of the file?

NO

- What is the HTTP status code and phrase returned from the server in response to the second HTTP GET? Did the server explicitly return the contents of the file?

304 NOT MODIFIED; NO

(HTTP CONDITIONAL REQUEST SCREENSHOT of the Postman GUI)

The screenshot shows the Postman interface for an HTTP GET request to `https://en.wikipedia.org/`. The 'Headers' tab is active, showing a single header: `If-Modified-Since: Mon, 28 Oct 2019 14:45:01 GMT`. The 'Body' tab is also visible. At the bottom, the response status is `304 Not Modified`, with a time of `82ms` and a size of `1.01 KB`. The 'Save Response' button is visible.

Q3. Retrieving Long Documents:

- How many HTTP GET request messages were sent by your browser?

1 request only

- What is the status code and phrase associated with the response to the HTTP GET request?

200 OK

http && tcp						
No.	Time	Source	Destination	Protocol	Length	Info
28	2.648470	192.168.1.6	158.106.138.13	HTTP	534	GET /big.txt HTTP/1.1
3804	33.820246	192.168.1.6	158.106.138.13	HTTP	465	GET /favicon.ico HTTP/1.1
3808	34.112145	158.106.138.13	192.168.1.6	HTTP	1512	HTTP/1.1 200 OK (image/x-icon)

```
▶ Frame 28: 534 bytes on wire (4272 bits), 534 bytes captured (4272 bits) on interface 0
▶ Ethernet II, Src: Apple_5e:89:4c (60:30:d4:5e:89:4c), Dst: Tp-LinkT_76:bc:f9 (34:e8:94:76:bc:f9)
▶ Internet Protocol Version 4, Src: 192.168.1.6, Dst: 158.106.138.13
▶ Transmission Control Protocol, Src Port: 58269, Dst Port: 80, Seq: 1, Ack: 1, Len: 468
▼ Hypertext Transfer Protocol
  ▶ GET /big.txt HTTP/1.1\r\n
    Host: norvig.com\r\n
    Connection: keep-alive\r\n
    Accept: */*\r\n
```

Nmap

Nmap ("Network Mapper") is a free and open source license utility for network discovery and security auditing. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime.

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. It was designed to rapidly scan large networks, but works fine against single hosts.

Nmap runs on all major computer operating systems, and official binary packages are available for Linux, Windows, and Mac OS X.

Nmap Features

- **Flexible:** Supports dozens of advanced techniques for mapping out networks filled with IP filters, firewalls, routers, and other obstacles. This includes many port scanning mechanisms (both TCP & UDP), OS detection, version detection, ping sweeps, and more.
- **Powerful:** Nmap has been used to scan huge networks of literally hundreds of thousands of machines.

- **Portable:** Most operating systems are supported, including Linux, Microsoft Windows, FreeBSD, OpenBSD, Solaris, IRIX, Mac OS X, HP-UX, NetBSD, Sun OS, Amiga, and more.
- **Easy:** While Nmap offers a rich set of advanced features for power users, you can start out as simply as "nmap -v -A *targethost*". Both traditional command line and graphical (GUI) versions are available to suit your preference. Binaries are available for those who do not wish to compile Nmap from source.
- **Free:** The primary goals of the Nmap Project is to help make the Internet a little more secure and to provide administrators/auditors/hackers with an advanced tool for exploring their networks. Nmap is available for free download, and also comes with full source code that you may modify and redistribute under the terms of the license.
- **Well Documented:** Significant effort has been put into comprehensive and up-to-date man pages, whitepapers, tutorials, and even a whole book.
- **Supported:** While Nmap comes with no warranty, it is well supported by a vibrant community of developers and users. Most of this interaction occurs on the Nmap mailing lists. Most bug reports and questions should be sent to the nmap-dev list, but only after you read the guidelines.
- **Acclaimed:** Nmap has won numerous awards, including "Information Security Product of the Year" by Linux Journal, Info World and Codetalker Digest. It has been featured in hundreds of magazine articles, several movies, dozens of books, and one comic book series.
- **Popular:** Thousands of people download Nmap every day, and it is included with many operating systems (Redhat Linux, Debian Linux, Gentoo, FreeBSD, OpenBSD, etc). It is among the top ten (out of 30,000) programs at the Freshmeat.Net repository. This is important because it lends Nmap its vibrant development and user support communities.

Nmap Commands

1. Detect details about hosts in the network

-sL (List Scan)

The list scan is a degenerate form of host discovery that simply lists each host of the network(s) specified, without sending any packets to the target hosts. By default, Nmap still does reverse-DNS resolution on the hosts to learn their names.

```
root@qikfreez: ~ x root@qikfreez: ~
root@qikfreez:~# nmap -v -sL 192.168.1.0/28
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:19 EDT
Initiating Parallel DNS resolution of 16 hosts. at 17:19
Completed Parallel DNS resolution of 16 hosts. at 17:19, 0.03s elapsed
Nmap scan report for 192.168.1.0
Nmap scan report for 192.168.1.1
Nmap scan report for 192.168.1.2
Nmap scan report for 192.168.1.3
Nmap scan report for 192.168.1.4
Nmap scan report for 192.168.1.5
Nmap scan report for 192.168.1.6
Nmap scan report for 192.168.1.7
Nmap scan report for 192.168.1.8
Nmap scan report for 192.168.1.9
Nmap scan report for 192.168.1.10
Nmap scan report for 192.168.1.11
Nmap scan report for 192.168.1.12
Nmap scan report for 192.168.1.13
Nmap scan report for 192.168.1.14
Nmap scan report for 192.168.1.15
Nmap done: 16 IP addresses (0 hosts up) scanned in 0.03 seconds
```

-sn (No port scan)

This option tells Nmap not to do a port scan after host discovery, and only print out the available hosts that responded to the host discovery probes. This is often known as a “ping scan”, but you can also request that traceroute and NSE host scripts be run.

```
root@qikfreez: ~ x root@qikfreez: ~
root@qikfreez:~# nmap -v -sn 192.168.1.0/28
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:15 EDT
Initiating ARP Ping Scan at 17:15
Scanning 7 hosts [1 port/host]
Completed ARP Ping Scan at 17:15, 0.21s elapsed (7 total hosts)
Initiating Parallel DNS resolution of 7 hosts. at 17:15
Completed Parallel DNS resolution of 7 hosts. at 17:15, 0.03s elapsed
Nmap scan report for 192.168.1.0 [host down]
Nmap scan report for 192.168.1.1
Host is up (0.0035s latency).
MAC Address: 34:E8:94:76:BC:F9 (Unknown)
Nmap scan report for 192.168.1.2 [host down]
Nmap scan report for 192.168.1.3
Host is up (0.042s latency).
MAC Address: C8:D7:79:94:C5:2B (Qingdao Haier TelecomLtd)
Nmap scan report for 192.168.1.4 [host down]
Nmap scan report for 192.168.1.6
Host is up (0.00034s latency).
MAC Address: 60:30:D4:5E:89:4C (Unknown)
Nmap scan report for 192.168.1.7 [host down]
Initiating Parallel DNS resolution of 1 host. at 17:15
Completed Parallel DNS resolution of 1 host. at 17:15, 0.01s elapsed
Nmap scan report for 192.168.1.5
Host is up.
Initiating Ping Scan at 17:15
Scanning 8 hosts [4 ports/host]
Completed Ping Scan at 17:15, 9.01s elapsed (8 total hosts)
Nmap scan report for 192.168.1.8 [host down]
Nmap scan report for 192.168.1.9 [host down]
Nmap scan report for 192.168.1.10 [host down]
Nmap scan report for 192.168.1.11 [host down]
Nmap scan report for 192.168.1.12 [host down]
Nmap scan report for 192.168.1.13 [host down]
Nmap scan report for 192.168.1.14 [host down]
Nmap scan report for 192.168.1.15 [host down]
Read data files from: /usr/bin/../share/nmap
Nmap done: 16 IP addresses (4 hosts up) scanned in 9.37 seconds
Raw packets sent: 75 (2.740KB) | Rcvd: 3 (84B)
```


2. Show all the open ports

--open

Only show open (or possibly open) ports

```
root@qikfreez:~# nmap --open google.com
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:33 EDT
Nmap scan report for google.com (172.217.160.174)
Host is up (0.015s latency).
Other addresses for google.com (not scanned): 2404:6800:4009:80a::200e
rDNS record for 172.217.160.174: bom05s12-in-f14.1e100.net
Not shown: 998 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 4.99 seconds
root@qikfreez:~#
```

3. Scan IP Portal

-sN; -sF; -sX (TCP NULL, FIN, and Xmas scans)

These three scan types exploit a subtle loophole in the TCP RFC to differentiate between open and closed ports.

```
root@qikfreez:~# nmap -sF google.com
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:51 EDT
Nmap scan report for google.com (172.217.160.174)
Host is up (0.014s latency).
Other addresses for google.com (not scanned): 2404:6800:4009:80a::200e
rDNS record for 172.217.160.174: bom05s12-in-f14.1e100.net
All 1000 scanned ports on google.com (172.217.160.174) are open|filtered

Nmap done: 1 IP address (1 host up) scanned in 4.11 seconds
root@qikfreez:~#
```

4. Find port ranges

-p <port ranges>

Only scan specific ports.

```
root@qikfreez:~# nmap -p 1-600 192.168.1.1
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:53 EDT
Nmap scan report for 192.168.1.1
Host is up (0.0038s latency).
Not shown: 597 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
80/tcp    open  http
MAC Address: 34:E8:94:76:BC:F9 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds
root@qikfreez:~#
```

5. Find protocol list

-sO

IP protocol scan allows you to determine which IP protocols (TCP, ICMP, IGMP, etc.) are supported by target machines.

```
root@qikfreez:~# nmap -sO google.com
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:35 EDT
Nmap scan report for google.com (172.217.160.174)
Host is up (0.015s latency).
Other addresses for google.com (not scanned): 2404:6800:4009:80a::200e
rDNS record for 172.217.160.174: bom05s12-in-f14.1e100.net
Not shown: 254 open|filtered protocols
PROTOCOL STATE SERVICE
1      open  icmp
6      open  tcp

Nmap done: 1 IP address (1 host up) scanned in 2.99 seconds
root@qikfreez:~# %
```

6. For virus and os detection

-O: Enable OS detection

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

```
root@qikfreez:~# nmap -O google.com
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-28 17:46 EDT
Nmap scan report for google.com (172.217.160.174)
Host is up (0.014s latency).
Other addresses for google.com (not scanned): 2404:6800:4009:80a::200e
rDNS record for 172.217.160.174: bom05s12-in-f14.1e100.net
Not shown: 998 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https
Warning: OSscan results may be unreliable because we could not find at least 1 open and 1
closed port
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
No OS matches for host

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.53 seconds
```

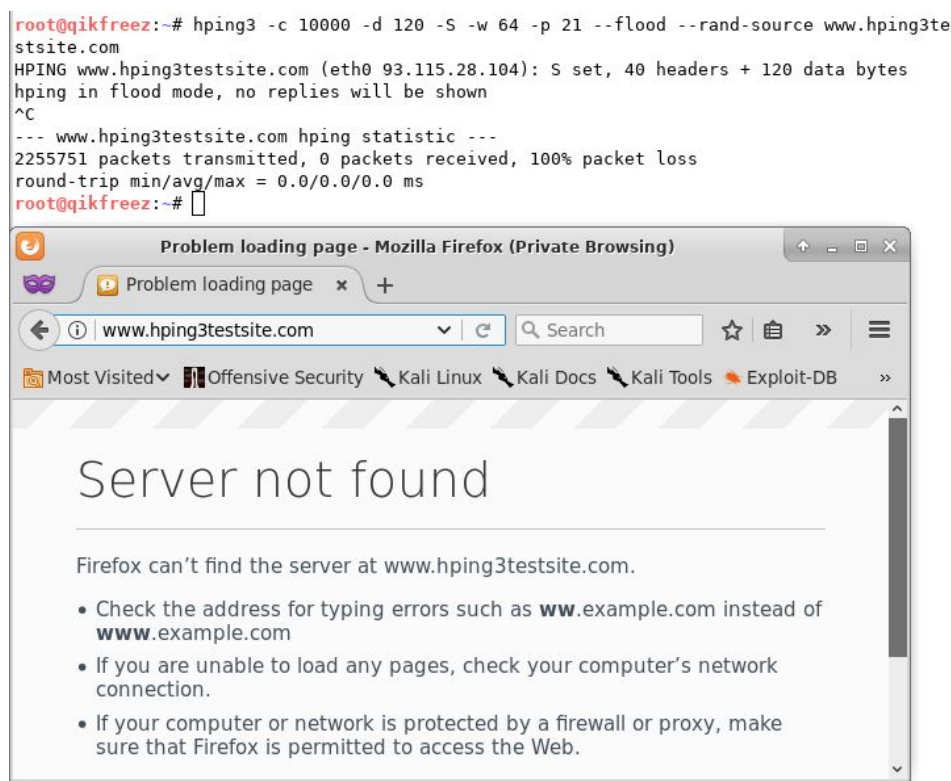

Hping3

hping3 is a network tool able to send custom TCP/IP packets and to display target replies like ping program does with ICMP replies. hping3 handle fragmentation, arbitrary packets body and size and can be used in order to transfer files encapsulated under supported protocols.

Hping3 Features:

- Test firewall rules
- Advanced port scanning
- Test net performance using different protocols, packet size, TOS (type of service) and fragmentation
- Path MTU discovery
- Transferring files between even really fascist firewall rules.
- Traceroute-like under different protocols.
- Firewalk-like usage
- Remote OS fingerprinting
- TCP/IP stack auditing.

Performing DOS attack using Hping3



Conclusion

We've explored multiple options and configurations of the tools Wireshark, Nmap & Hping3 and analyzed public and private network environments.