



# **Computer Networks**

**Phase 1 - Web Server**

**Projeto ISEL 2023/24 — LEETC**

## **Coordination**

General: Carlos Meneses      Course: Nuno Cruz

## **Grupo LP-07**

Supervisor: Luís Pires

## **Student**

Nuno Brito <A46948@alunos.isel.pt>

April 14th 2024

# Contents

<b>Figure list</b>	<b>ii</b>
<b>Table list</b>	<b>iii</b>
<b>Listings list</b>	<b>iv</b>
<b>Acronyms list</b>	<b>v</b>
<b>Glossary</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Phase 1</b>	<b>2</b>
2.1 Milestones . . . . .	2
2.2 WebClient requirements . . . . .	2
2.3 Software . . . . .	2
2.4 Software install . . . . .	3
2.5 WebClient - Python Code . . . . .	6
2.6 Wireshark captures . . . . .	9
2.7 List of headers and replies . . . . .	14
<b>3 Phase 2</b>	<b>16</b>
3.1 Connecting two devices with a switch . . . . .	16
3.1.1 Simulating a network using Cisco Packet Tracer . . . . .	16
3.2 Connecting two LANs with a router . . . . .	19
<b>4 Issues and fixes</b>	<b>36</b>
<b>5 Conclusions</b>	<b>37</b>
<b>A Appendix</b>	<b>38</b>

# List of Figures

2.1	Changing from false to true the <i>security.tls.version.enable-deprecated</i> option . . . . .	10
2.2	Webclient get capture . . . . .	10
2.3	Browser capture . . . . .	11
2.4	Webclient reply capture . . . . .	11

# List of Tables

2.1 XAMPP install . . . . .	5
2.2 Wireshark install . . . . .	6
3.1 Cisco Packet Tracer guide . . . . .	18
3.2 Visual LAN allocation . . . . .	20
3.3 LAN allocation table . . . . .	20
3.4 IP configuration table . . . . .	21
3.5 Cisco Packet Tracer guide . . . . .	26

# Listings

2.1	Simple HTTP WebClient using sockets in python . . . . .	6
2.2	WebClient output . . . . .	7
2.3	Wireshark capture output sample - VPN . . . . .	12
2.4	Wireshark capture output sample - Browser . . . . .	13
2.5	Wireshark capture output sample - WebClient . . . . .	14
3.1	PC0 CMD output . . . . .	18
3.2	Laptop0 CMD output . . . . .	19
3.3	Network plan . . . . .	19
3.4	PC0 CMD output . . . . .	26
3.5	Laptop0 CMD output . . . . .	27
3.6	PC1 CMD output . . . . .	29
3.7	Laptop1 CMD output . . . . .	31
3.8	Router 1 CLI output . . . . .	33

# Acronyms list

API	Application Programming Interface
CLI	Command Line Interface
CMD	Command Line Prompt
GUI	Graphical User Interface
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
IP	Internet Protocol
LAN	Local Area Network
OS	Operating System
OSS	openSUSE
PC	Personal Computer
PHP	PHP: Hypertext Preprocessor
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TUI	Terminal User Interface
UDP	User Datagram Protocol
VPN	Virtual Private Network
WWW	World Wide Web
XAMPP	Cross-Platform, Apache, MySQL, PHP, and Perl

# Glossary

## **Apache2**

An opensource HTTP web server.

## **Broadcast**

## **Browser**

A browser is a internet navigation software. It comes in multiple flavours, nowadays the big three are Microsoft Edge, Mozilla Firefox and Google Chrome.

## **Cisco Packet Tracer**

## **Command Line Prompt**

## **Firewall**

A barrier between networks. Controls inbound and outbound traffic.

## **Gateway**

## **LibreWolf**

An internet browser based on Mozilla's Firefox. Its primary purpose is to allow privacy, and with it comes security. It achieves this by removing telemetry and data collection.

## **Linux**

## **MariaDB**

A community-developed fork of MySQL database server.

## **Network**

## **openSUSE Tumbleweed**

An openSUSE (OSS) is an open-source community driven Linux-based distribution sponsored by SUSE Software Solutions. Tumbleweed is a rolling release version allowing for up-to-date software releases.

## **Operating system**

A program that manages a computer's resources from software to hardware.

## **Ping**

## **Tracert**

## **Ipconfig**

## **Python**

Python is a high-level programming language, object-oriented.

## **Perl**

A high-level, general-purpose, interpreted, dynamic programming language

## **Rolling release distribution**

A distribution where its software release cycle is more frequent than those of Long Term Support (LTS). It's up to the Linux-based distributor to guarantee the testing of a package.

## **Router**

## **Switch**

## **Socket**

A network socket serves as an endpoint for sending and receiving data across the network.

## **Subnet Mask**

## **Unix**

## **VPN**

A private network creating a secure connection between a device and a network.

## **Windows**

Microsoft's operating system. First released in 1985 as a Graphical User Interface (GUI) for MS-DOS, continued to evolve with its latest version being 11. Due to its nature, it's not recommended for server production environment.

## **Wireshark**

Wireshark is a network protocol analyser software. Allows traffic capture between a computer and a network.

## **XAMPP**

A software package environment collection containing Apache2 webserver, MariaDB database, PHP and Perl.

# **Chapter 1**

## **Introduction**

The project consists in building a computer network through four phases. First with a webserver, then simulating a local area network (LAN) with two computers and a switch. By the end of the journey, this project will develop into something similar to a corporate network.

# **Chapter 2**

## **Phase 1**

### **2.1 Milestones**

- Setup apache2 web server in localhost
- Access web server locally (<http://127.0.0.1/> or <http://localhost>)
- Access web server from a remote computer (<http://172.24.1.12>)
- Use wireshark in a remote host to capture packages from the server
- Compare the HTTP headers sent by the client and the server
- Develop a simple barebones HTTP webclient
- Establish a TCP connection to the server
- Request the base webpage

### **2.2 WebClient requirements**

- HTTP library forbidden
- Establish TCP connection using available sockets library - send/receive the HTTP request/reply
- Output HTTP reply to the user
- - Optional - act to the various HTTP replies
- Text-only application

### **2.3 Software**

- Local server side
  - Operating system: Windows 11 x64
  - WebServer: XAMPP x64 8.2.12-0-VS16 for windows

- Client side

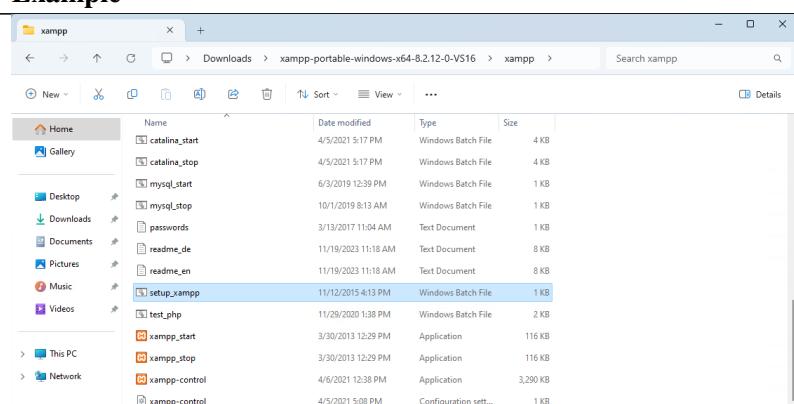
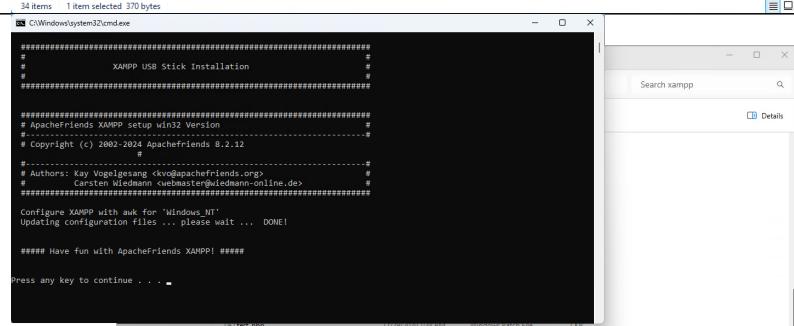
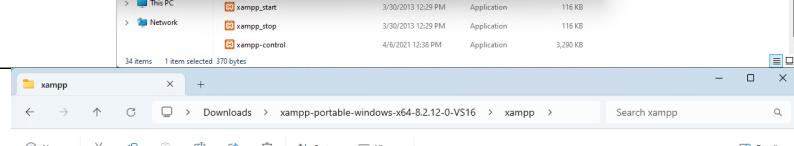
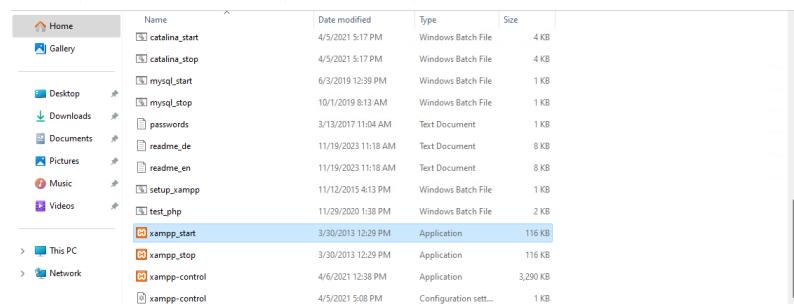
Operating system: openSUSE Tumbleweed

Browser: LibreWolf version 123.0-1

Package monitor: Wireshark version 4.2.3 (Git commit b0da86c196d1).

## 2.4 Software install

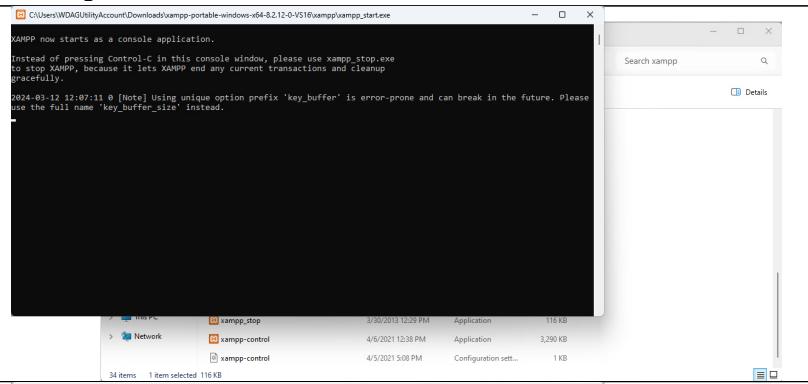
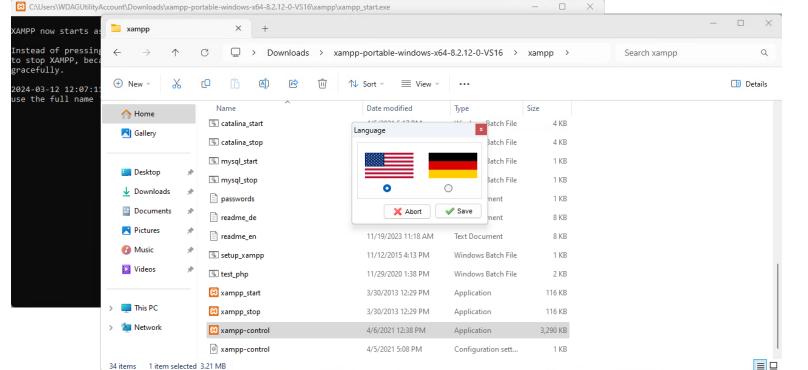
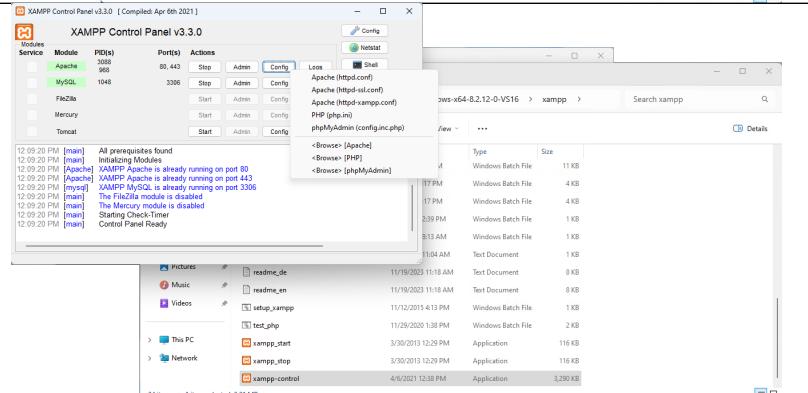
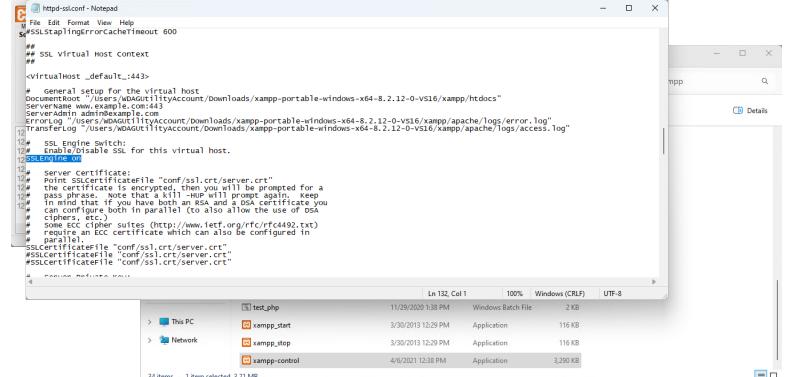
First we install, start and configure XAMPP. Using the following link, <https://www.apachefriends.org/download.html>, we can choose our preferred method, for this project the portable version was the best choice since no installation was needed. After uncompressed our downloaded file, we can start the process.

Steps	Example
The read me file available in the root directory states that we need to run <code>setup_xampp</code> batch file first to populate the registry it's directory.	
After completing, just press any button to continue.	
Next click in the <code>xampp_start</code> executable, windows will prompt some firewall permissions which will gladly accept.	
Continued on next page	

**Table 2.1 continued from previous page**

**Steps**

**Example**

<p>Pick a language for the program.</p> 	
<p>Apache2 comes with SSL on by default. Since we're studying the HTTP protocol it's convenient to disable HTTPS. Pressing the config button shows a list of configuration files. Select <i>Apache</i> (<i>httpd-ssl.conf</i>).</p> 	
<p>Search and comment the line with <i>SSLEngine on</i>.</p> 	

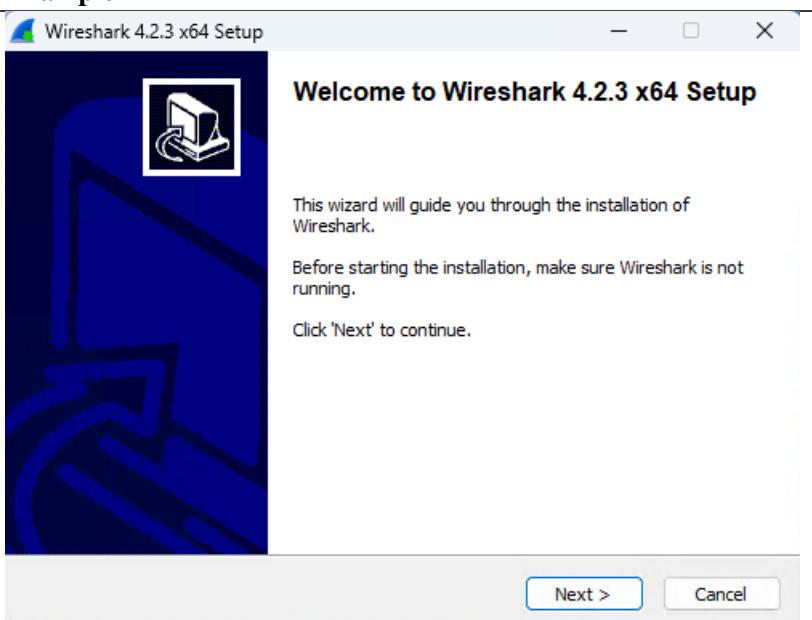
Continued on next page

**Table 2.1 continued from previous page**

Steps	Example
Previously if we went to <code>http://localhost</code> it would redirect to the HTTPS version. After completing the above step it'll no longer redirect, showing us the non-secure version.	

Table 2.1: XAMPP install

Next up is wireshark, the powerful network analyser. We'll download the installer from <https://www.wireshark.org/download.html>.

Steps	Example
Wireshark setup. Select everything available and continue by clicking <i>next</i> .	

Continued on next page

**Table 2.2 continued from previous page**

Steps	Example
After completing the installation, reboot the computer.	

Table 2.2: Wireshark install

## 2.5 WebClient - Python Code

```
1 """
2 Name: Python TCP Client
3 Description: Simple TCP client using sockets
4 Original code by: Luis Pires
5 Source: Chapter 2, slide 104
6
7 Commented and adapted by: Nuno Brito
8 """
9
10 # Import from everything from the socket library
11 from socket import *
12
13 # Specify servername and port destination
14 serverName = "172.24.1.12"
15 serverPort = 80
16
17 # GET list
18 httpTestMessages = [
19     "GET /dashboard/ HTTP/1.1\r\n",
20     "GET /dashboard HTTP/1.1\r\n",
21     "PUT / HTTP/1.1\r\n",
22     "GET /dashboard HTTP/1.\r\n",
23     "GET /dashboard/index.htm HTTP/1.1\r\n",
24     "PUT /d HTTP/1.1\r\n",
25     "BREW /coffee/ HTTP/1.1\r\n",
26 ]
27
28 # Cycle through predefined messages
```

```

29 for sentence in httpTestMessages:
30
31     # Socket open and connect
32     clientSocket = socket(AF_INET, SOCK_STREAM)
33     clientSocket.connect((serverName, serverPort))
34
35     # Join serverName to the current sentence
36     sentence += "Host:" + serverName + "\r\n\r\n"
37
38     # Socket encode message and send
39     clientSocket.send(sentence.encode())
40
41     # Receive and out the response message
42     modifiedSentence = clientSocket.recv(1024)
43
44     # Close socket connection
45     clientSocket.close()
46
47     # Print the requested message
48     print ("-"*60)
49     print ("From Server:", modifiedSentence.decode())

```

Listing 2.1: Simple HTTP WebClient using sockets in python

The code listed in 2.1 was adapted to be simple and cycle through the various request messages without any user input.

Modifiable variables include *serverName*, *serverPort* and the *httpTestMessages* list.

The webclient produces the following output:

```

1 -----
2 From Server: HTTP/1.1 200 OK
3 Date: Fri, 29 Mar 2024 18:42:08 GMT
4 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
5 Last-Modified: Sun, 19 Nov 2023 11:18:30 GMT
6 ETag: "1443-60a7f87754d80"
7 Accept-Ranges: bytes
8 Content-Length: 5187
9 Content-Type: text/html
10
11 <!doctype html>
12 <html lang="en">
13     <head>
14         <meta charset="utf-8">
15         <!-- Always force latest IE rendering engine or request Chrome Frame -->
16         <meta content="IE=edge,chrome=1" http-equiv="X-UA-Compatible">
17         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
18
19         <!-- Use title if it's in the page YAML frontmatter -->
20         <title>Welcome to XAMPP</title>
21
22         <meta name="description" content="XAMPP is an easy to install Apache
23             distribution containing MariaDB, PHP and Perl." />
24         <meta name="keywords" content="xampp, apache, php, perl, mariadb, open source
25             distribution" />
26
27         <link href="/dashboard/stylesheets/normalize.css" rel="stylesheet" type="text/
28             css" /><link href="/dashboard/stylesheets/all.css" rel="stylesheet" type="t
29 -----
30 From Server: HTTP/1.1 301 Moved Permanently

```

```

28 Date: Fri, 29 Mar 2024 18:42:08 GMT
29 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
30 Location: http://172.24.1.12/dashboard/
31 Content-Length: 338
32 Content-Type: text/html; charset=iso-8859-1
33
34 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
35 <html><head>
36 <title>301 Moved Permanently</title>
37 </head><body>
38 <h1>Moved Permanently</h1>
39 <p>The document has moved <a href="http://172.24.1.12/dashboard/">here</a>.</p>
40 <hr>
41 <address>Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at 172.24.1.12 Port
     80</address>
42 </body></html>
43 -----
44 From Server: HTTP/1.1 302 Found
45 Date: Fri, 29 Mar 2024 18:42:08 GMT
46 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
47 X-Powered-By: PHP/8.2.12
48 Location: http://172.24.1.12/dashboard/
49 Content-Length: 0
50 Content-Type: text/html; charset=UTF-8
51 -----
52 From Server: HTTP/1.1 400 Bad Request
53 Date: Fri, 29 Mar 2024 18:42:08 GMT
54 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
55 Content-Length: 325
56 Connection: close
57 Content-Type: text/html; charset=iso-8859-1
58
59 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
60 <html><head>
61 <title>400 Bad Request</title>
62 </head><body>
63 <h1>Bad Request</h1>
64 <p>Your browser sent a request that this server could not understand.<br />
65 </p>
66 <hr>
67 <address>Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at localhost Port
     80</address>
68 </body></html>
69 -----
70 From Server: HTTP/1.1 404 Not Found
71 Date: Fri, 29 Mar 2024 18:42:08 GMT
72 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
73 Content-Length: 297
74 Content-Type: text/html; charset=iso-8859-1
75
76 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
77 <html><head>
78 <title>404 Not Found</title>
79 </head><body>
80 <h1>Not Found</h1>
81 <p>The requested URL was not found on this server.</p>
82 <hr>
83 <address>Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at 172.24.1.12 Port
     80</address>
84 </body></html>
85 -----
86 From Server: HTTP/1.1 405 Method Not Allowed
87 Date: Fri, 29 Mar 2024 18:42:08 GMT

```

```

88 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
89 Allow: HEAD,GET,POST,OPTIONS,TRACE
90 Content-Length: 321
91 Content-Type: text/html; charset=iso-8859-1
92
93 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
94 <html><head>
95 <title>405 Method Not Allowed</title>
96 </head><body>
97 <h1>Method Not Allowed</h1>
98 <p>The requested method PUT is not allowed for this URL.</p>
99 <hr>
100 <address>Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at 172.24.1.12 Port
     80</address>
101 </body></html>
102 -----
103 From Server: HTTP/1.1 501 Not Implemented
104 Date: Fri, 29 Mar 2024 18:42:08 GMT
105 Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
106 Allow: HEAD,GET,POST,OPTIONS,TRACE
107 Content-Length: 304
108 Connection: close
109 Content-Type: text/html; charset=iso-8859-1
110
111 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
112 <html><head>
113 <title>501 Not Implemented</title>
114 </head><body>
115 <h1>Not Implemented</h1>
116 <p>BREW not supported for current URL.<br />
117 </p>
118 <hr>
119 <address>Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at 172.24.1.12 Port
     80</address>
120 </body></html>

```

Listing 2.2: WebClient output

## 2.6 Wireshark captures

First we must ensure the browser being used can connect to the XAMPP server with HTTP. We can do that by enabling the usage of deprecated TLS.

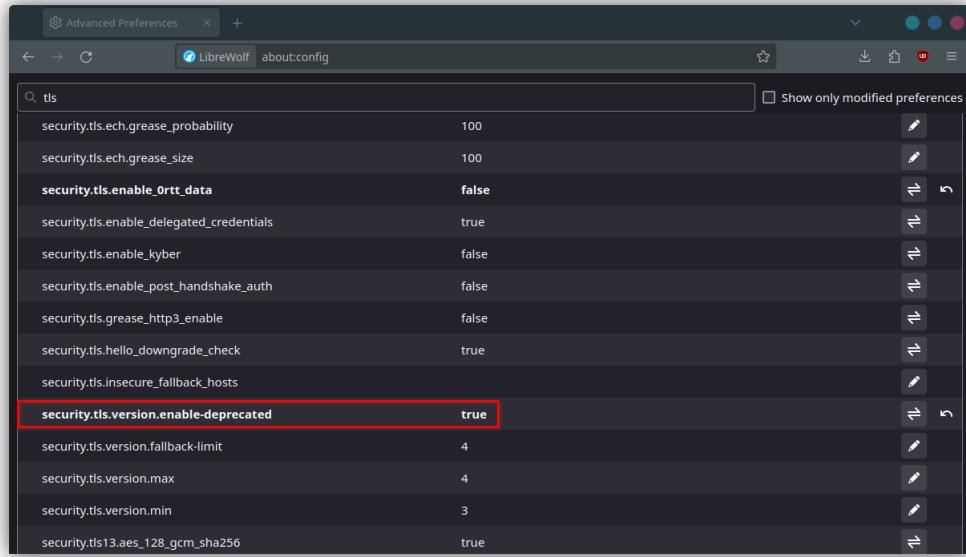


Figure 2.1: Changing from false to true the *security.tls.version.enable-deprecated* option

Then we can start our capture process, next follows some printscreens examples filtered by HTTP.

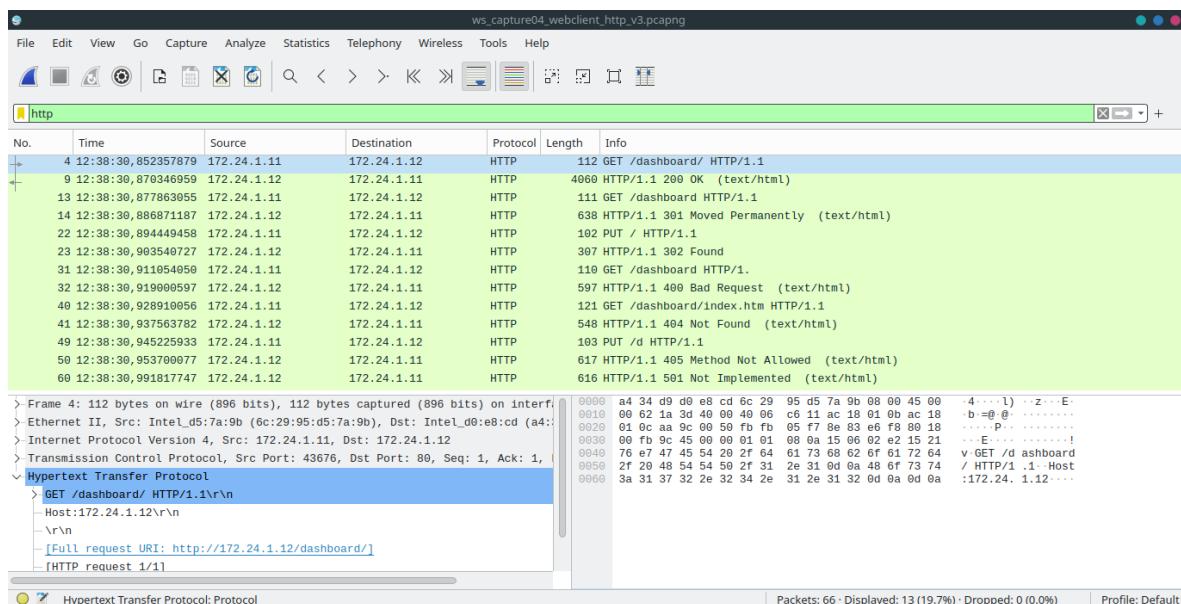


Figure 2.2: Webclient get capture

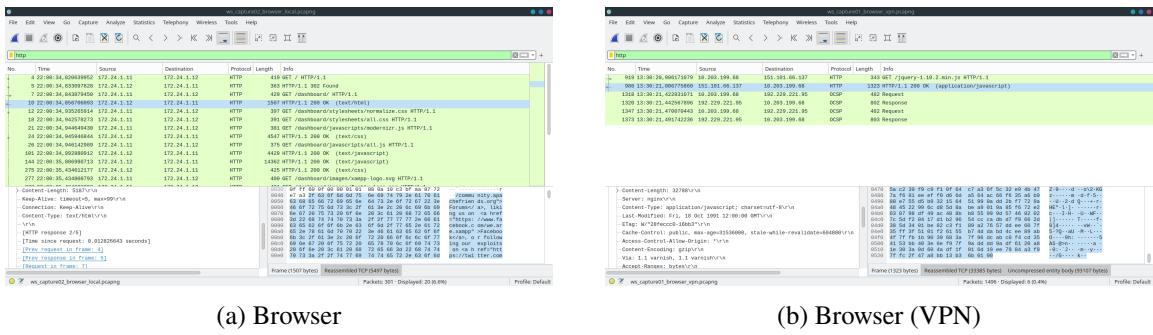


Figure 2.3: Browser capture

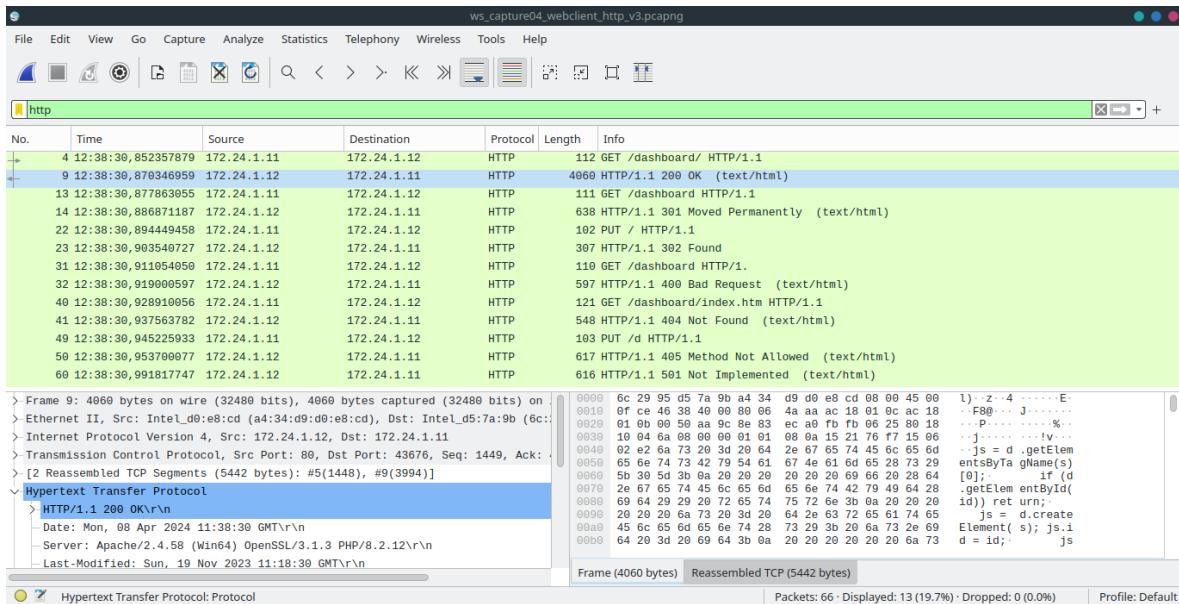


Figure 2.4: Webclient reply capture

To compliment the images, below are segmented outputs (the important parts):

No.	Time	Source	Destination	Protocol
1	Length Info			
2	919 13:30:20,986171079	10.203.199.68	151.101.66.137	HTTP 343
3	GET /jquery-1.10.2.min.js	HTTP/1.1		
4	Hypertext Transfer Protocol			
5	GET /jquery-1.10.2.min.js	HTTP/1.1\r\n		
6	[Expert Info (Chat/Sequence): GET /jquery-1.10.2.min.js	HTTP/1.1\r\n]		
7	[GET /jquery-1.10.2.min.js	HTTP/1.1\r\n]		
8	[Severity level: Chat]			
9	[Group: Sequence]			
10	Request Method: GET			
11	Request URI: /jquery-1.10.2.min.js			
12	Request Version: HTTP/1.1			
13	Host: code.jquery.com\r\n			
14	User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:123.0) Gecko/20100101 Firefox/123.0\r\n			
15	Accept: */*\r\n			
16	Accept-Language: en-US,en;q=0.5\r\n			
17	Accept-Encoding: gzip, deflate\r\n			
18	Connection: keep-alive\r\n			
19	Referer: http://172.24.1.12/\r\n			
20	\r\n			
21	[Full request URI: http://code.jquery.com/jquery-1.10.2.min.js]			
22	[HTTP request 1/1]			
23	[Response in frame: 986]			
No.	Time	Source	Destination	Protocol
24	Length Info			
25	986 13:30:21,006775860	151.101.66.137	10.203.199.68	HTTP 1323
26	HTTP/1.1 200 OK (application/javascript)			
27	Hypertext Transfer Protocol			
28	HTTP/1.1 200 OK\r\n			
29	[Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]			
30	[HTTP/1.1 200 OK\r\n]			
31	[Severity level: Chat]			
32	[Group: Sequence]			
33	Response Version: HTTP/1.1			
34	Status Code: 200			
35	[Status Code Description: OK]			
36	Response Phrase: OK			
37	Connection: keep-alive\r\n			
38	Content-Length: 32788\r\n			
39	[Content length: 32788]			
40	Server: nginx\r\n			
41	Content-Type: application/javascript; charset=utf-8\r\n			
42	Last-Modified: Fri, 18 Oct 1991 12:00:00 GMT\r\n			
43	ETag: W/"28fecccc0-16bb3"\r\n			
44	Cache-Control: public, max-age=31536000, stale-while-revalidate=604800\r\n			
45	Access-Control-Allow-Origin: *\r\n			
46	Content-Encoding: gzip\r\n			
47	Via: 1.1 varnish, 1.1 varnish\r\n			
48	Accept-Ranges: bytes\r\n			
49	Date: Mon, 11 Mar 2024 13:30:20 GMT\r\n			
50	Age: 15363215\r\n			
51	X-Served-By: cache-lga13622-LGA, cache-lis1490024-LIS\r\n			
52	X-Cache: HIT, HIT\r\n			
53	X-Cache-Hits: 12, 7599\r\n			
54	X-Timer: S1710163821.991495,VS0,VE0\r\n			
55	Vary: Accept-Encoding\r\n			
	\r\n			
	[HTTP response 1/1]			

```

56 [Time since request: 0.020604781 seconds]
57 [Request in frame: 919]
58 [Request URI: http://code.jquery.com/jquery-1.10.2.min.js]
59 Content-encoded entity body (gzip): 32788 bytes -> 93107 bytes
60 File Data: 93107 bytes
61 Media Type
62 Media type: application/javascript; charset=utf-8 (93107 bytes)

```

Listing 2.3: Wireshark capture output sample - VPN

No.	Time	Source	Destination	Protocol	
1	Length Info				
2	4 22:00:34,820639952	172.24.1.11	172.24.1.12	HTTP 419	
	GET / HTTP/1.1				
3	Hypertext Transfer Protocol				
4	GET / HTTP/1.1\r\n				
5	[Expert Info (Chat/Sequence): GET / HTTP/1.1\r\n]				
6	[GET / HTTP/1.1\r\n]				
7	[Severity level: Chat]				
8	[Group: Sequence]				
9	Request Method: GET				
10	Request URI: /				
11	Request Version: HTTP/1.1				
12	Host: 172.24.1.12\r\n				
13	User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:123.0) Gecko/20100101 Firefox/123.0\r\n				
14	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8\r\n				
15	Accept-Language: en-US,en;q=0.5\r\n				
16	Accept-Encoding: gzip, deflate\r\n				
17	DNT: 1\r\n				
18	Sec-GPC: 1\r\n				
19	Connection: keep-alive\r\n				
20	Upgrade-Insecure-Requests: 1\r\n				
21	\r\n				
22	[Full request URI: http://172.24.1.12/]				
23	[HTTP request 1/5]				
24	[Response in frame: 5]				
25	[Next request in frame: 7]				
26	No.	Time	Source	Destination	Protocol
	Length Info				
27	5 22:00:34,833097828	172.24.1.12	172.24.1.11	HTTP 363	
	HTTP/1.1 302 Found				
28	Hypertext Transfer Protocol				
29	HTTP/1.1 302 Found\r\n				
30	[Expert Info (Chat/Sequence): HTTP/1.1 302 Found\r\n]				
31	[HTTP/1.1 302 Found\r\n]				
32	[Severity level: Chat]				
33	[Group: Sequence]				
34	Response Version: HTTP/1.1				
35	Status Code: 302				
36	[Status Code Description: Found]				
37	Response Phrase: Found				
38	Date: Tue, 12 Mar 2024 22:00:35 GMT\r\n				
39	Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12\r\n				
40	X-Powered-By: PHP/8.2.12\r\n				
41	Location: http://172.24.1.12/dashboard/\r\n				
42	Content-Length: 0\r\n				
43	[Content length: 0]				
44	Keep-Alive: timeout=5, max=100\r\n				
45	Connection: Keep-Alive\r\n				

```

46 Content-Type: text/html; charset=UTF-8\r\n
47 \r\n
48 [HTTP response 1/5]
49 [Time since request: 0.012457876 seconds]
50 [Request in frame: 4]
51 [Next request in frame: 7]
52 [Next response in frame: 10]
53 [Request URI: http://172.24.1.12/]

```

Listing 2.4: Wireshark capture output sample - Browser

No.	Time	Source	Destination	Protocol	
1	Length Info				
2	4 12:38:30,852357879	172.24.1.11	172.24.1.12	HTTP 112	
	GET /dashboard/ HTTP/1.1				
3	Hypertext Transfer Protocol				
4	No.	Time	Source	Destination	Protocol
	Length Info				
5	9 12:38:30,870346959	172.24.1.12	172.24.1.11	HTTP 4060	
	HTTP/1.1 200 OK (text/html)				
6	Hypertext Transfer Protocol				
	HTTP/1.1 200 OK\r\n				
7	Date: Mon, 08 Apr 2024 11:38:30 GMT\r\n				
8	Server: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12\r\n				
9	Last-Modified: Sun, 19 Nov 2023 11:18:30 GMT\r\n				
10	ETag: "1443-60a7f87754d80"\r\n				
11	Accept-Ranges: bytes\r\n				
12	Content-Length: 5187\r\n				
13	Content-Type: text/html\r\n				
14	\r\n				
15	[HTTP response 1/1]				
16	[Time since request: 0.017989080 seconds]				
17	[Request in frame: 4]				
18	[Request URI: http://172.24.1.12/dashboard/]				
19	File Data: 5187 bytes				
20	Line-based text data: text/html (130 lines)				

Listing 2.5: Wireshark capture output sample - WebClient

## 2.7 List of headers and replies

**Request:** GET /dashboard/ HTTP/1.1\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 200 OK

Meaning: this header complies with what the server expects from a webclient request.

**Request:** GET /dashboard HTTP/1.1\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 301 Moved Permanently

Meaning: this header request a relative directory without a forward slash at the end, prompting the server to reply with a "moved" answer.

**Request:** PUT / HTTP/1.1\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 302 Found

Meaning: this header request is an upload request to an unexistent directory.

**Request:** GET /dashboard HTTP/1.\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 400 Bad Request

Meaning: this header request, although it has an invalid directory, has the HTTP protocol version badly written (HTTP/1.1 vs. actual HTTP/1.) which causes a "bad request" reply from the server.

**Request:** GET /dashboard/index.htm HTTP/1.1\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 404 Not Found

Meaning: this header request tries to get a file that doesn't exist in the local server.

**Request:** PUT /d HTTP/1.1\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 405 Method Not Allowed

Meaning: this header request tries to upload something to the relative directory "d".

**Request:** BREW /coffee/ HTTP/1.1\r\nHost:127.24.1.12 \r\n\r\n

**Reply:** HTTP/1.1 501 Not Implemented

Meaning: a poorly attempt to get the 1998 April fool's day. It should have replied with 418 I'm a teapot. Even with GET instead of BREW it didn't work. Apache doesn't have the implementation.

# Chapter 3

## Phase 2

### 3.1 Connecting two devices with a switch

This first part is very simple. There are two devices (PC0 and Laptop0) connected to a switch and their network starts with 192.168.**GROUP NUMBER**.0.

Therefore:

- Group: 7 [192.168.7.0/24]
- Laptop0 [192.168.7.1]
- PC0 [192.168.7.2]

After applying the configuration we must run a set of commands to test our network.

- Ping: to test connectivity between devices over IP.
- Tracert: diagnostic command for displaying possible routes, also measures transit delay of packages across IP.
- Ipconfig: console application program of some computer operating systems that displays all current TCP/IP network configuration values. Unix and linux equivalent is *ifconfig*.

#### 3.1.1 Simulating a network using Cisco Packet Tracer

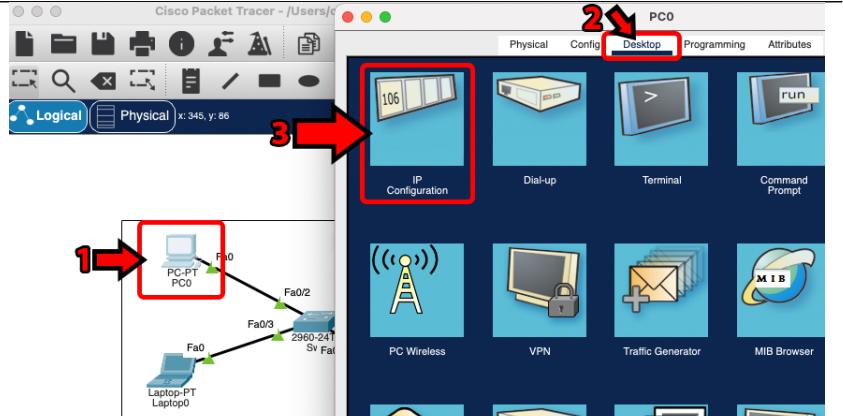
For this project Cisco Packet Tracer will be our main tool. Using the Command Line Prompt (CMD) in each device, will simulate all referenced commands in the network.

So let's get in this magnificant world starting with these steps:

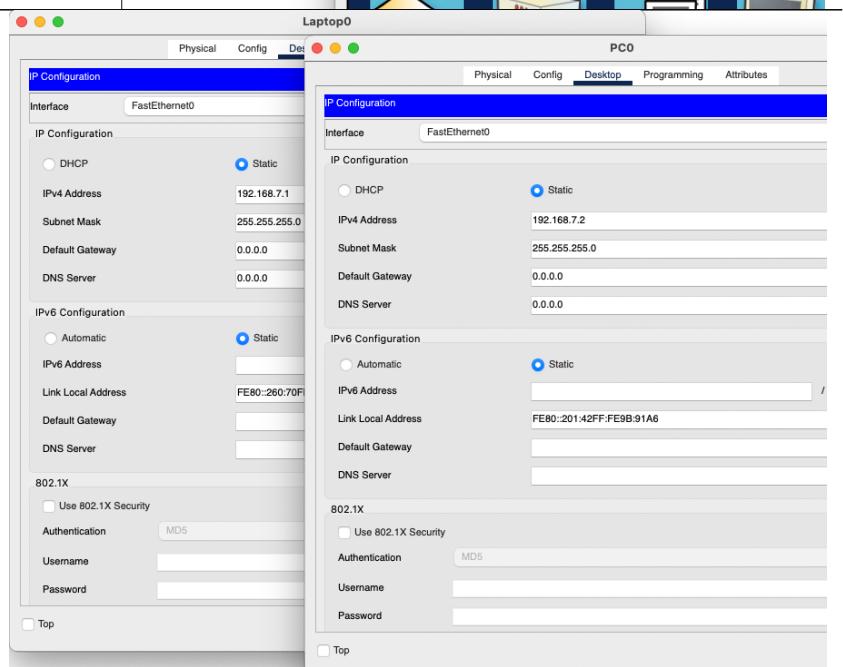
## Steps

To configure the IP on a device we must **single-click** in the intended device (1), go to the *desktop tab* (2) and select *IP Configuration* (3).

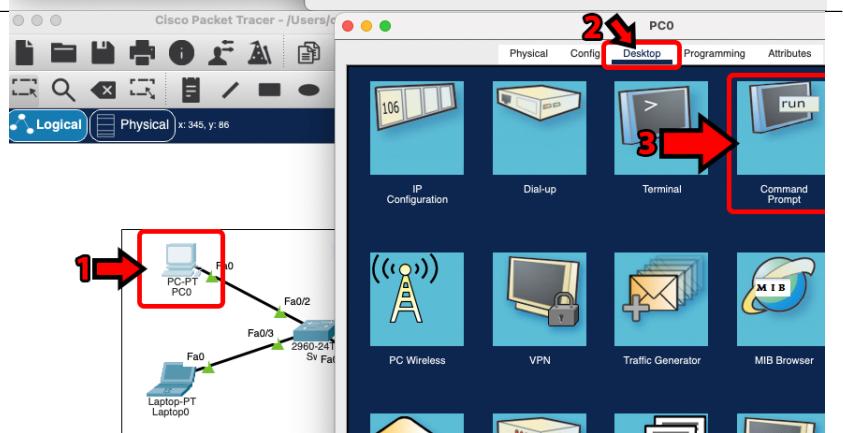
## Example



Now we configure our devices according to the required addresses.



To test the connection we select a device by **single-clicking** it (1), go to the *desktop tab* (2) and select *Command Prompt* (3).

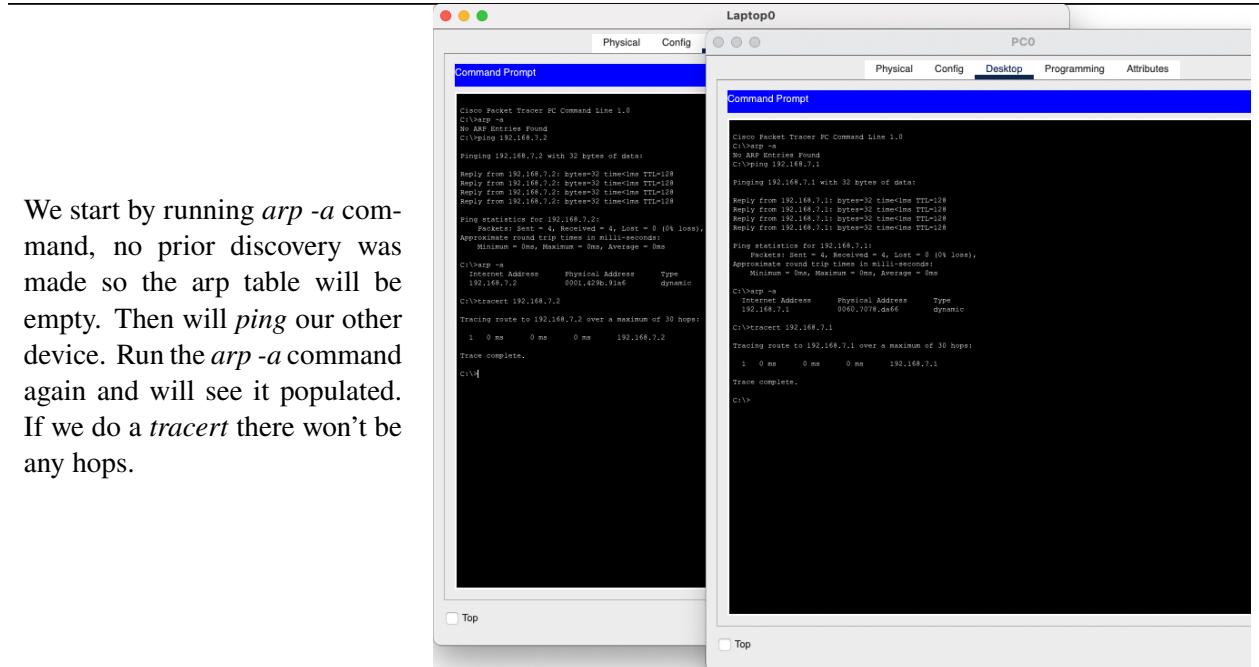


Continued on next page

**Table 3.1 continued from previous page**

**Steps**

**Example**



**Table 3.1: Cisco Packet Tracer guide**

And to supplement the output images, below are the text versions:

```

1 Cisco Packet Tracer PC Command Line 1.0
2 C:>arp -a
3 No ARP Entries Found
4 C:>ping 192.168.7.1
5
6 Pinging 192.168.7.1 with 32 bytes of data:
7
8 Reply from 192.168.7.1: bytes=32 time<1ms TTL=128
9 Reply from 192.168.7.1: bytes=32 time<1ms TTL=128
10 Reply from 192.168.7.1: bytes=32 time<1ms TTL=128
11 Reply from 192.168.7.1: bytes=32 time<1ms TTL=128
12
13 Ping statistics for 192.168.7.1:
14     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
15 Approximate round trip times in milli-seconds:
16     Minimum = 0ms, Maximum = 0ms, Average = 0ms
17
18 C:>arp -a
19     Internet Address          Physical Address          Type
20         192.168.7.1           00E0.7078.da66        dynamic
21
22 C:>tracert 192.168.7.1
23
24 Tracing route to 192.168.7.1 over a maximum of 30 hops:
25
26     1  0 ms      0 ms      0 ms      192.168.7.1
27
28 Trace complete.
29
30 C:>

```

**Listing 3.1: PC0 CMD output**

---

```

1 Cisco Packet Tracer PC Command Line 1.0
2 C:>arp -a
3 No ARP Entries Found
4 C:>ping 192.168.7.2
5
6
7 Pinging 192.168.7.2 with 32 bytes of data:
8
9 Reply from 192.168.7.2: bytes=32 time<1ms TTL=128
10 Reply from 192.168.7.2: bytes=32 time<1ms TTL=128
11 Reply from 192.168.7.2: bytes=32 time<1ms TTL=128
12 Reply from 192.168.7.2: bytes=32 time<1ms TTL=128
13
14 Ping statistics for 192.168.7.2:
15     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
16 Approximate round trip times in milli-seconds:
17     Minimum = 0ms, Maximum = 0ms, Average = 0ms
18
19 C:>arp -a
20     Internet Address          Physical Address          Type
21         192.168.7.2           0001.429b.91a6        dynamic
22
23 C:>tracert 192.168.7.2
24
25 Tracing route to 192.168.7.2 over a maximum of 30 hops:
26
27     1  0 ms      0 ms      0 ms      192.168.7.2
28
29 Trace complete.
30
31 C:>

```

---

Listing 3.2: Laptop0 CMD output

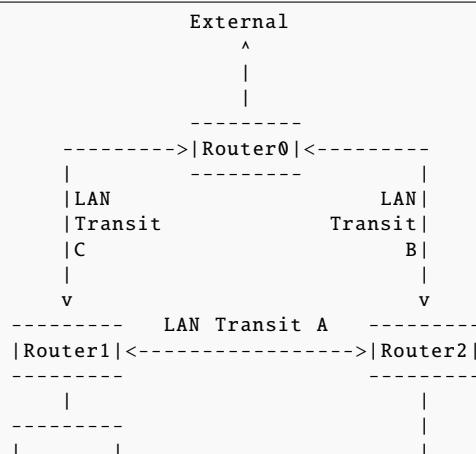
**Question:** How can a PC know if it is connected to a switch? Is traceroute useful in this situation?

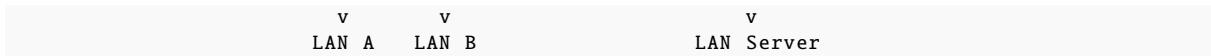
**Answer:**

If a device is connected to a switch, the arp table will include both devices IP addresses after a ping to each other. However, if connected to a router (as will see in the next section), they will only include their gateways IP addresses.

Traceroute isn't very useful here. It only shows hops in a routed network path (layer 3). Layer 2 devices such as switches won't show up since they receive and forward ethernet frames.

## 3.2 Connecting two LANs with a router





Listing 3.3: Network plan

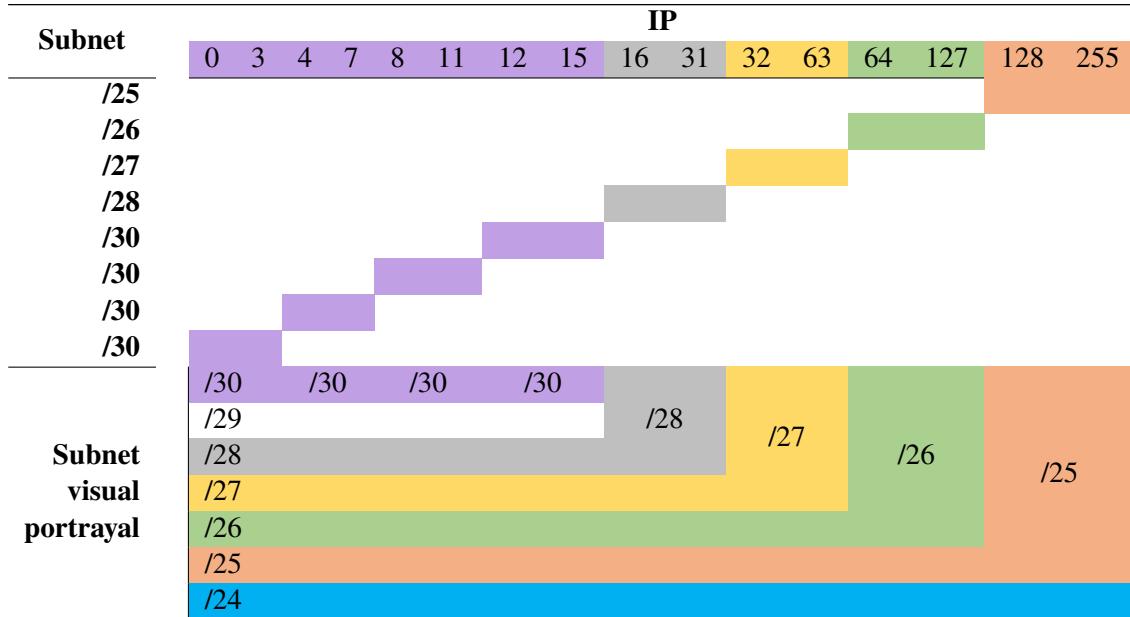


Table 3.2: Visual LAN allocation

Name	Network	Usable IPs	Router	Broadcast	Subnet Mask	Populated
192.168.7.						
LAN Server	128	129 - 253	254	255	128	126
LAN A	64	65 - 125	126	127	192	48
LAN B	32	33 - 61	62	63	224	27
Unused remaining	16	17 - 31		32		0
	12	13 - 14		15		0
LAN Transit C	8	9 - 10		11	252	2
LAN Transit B	4	5 - 6		7	252	2
LAN Transit A	0	1 - 2		3	252	2

Table 3.3: LAN allocation table

Name	Ports Link	Network	IP	Subnet Mask	Gateway
PC0	Fa0 - Sw0 Fa0/2	LAN A	192.168.7.65	255.255.255.192	192.168.7.126
Laptop0	Fa0 - Sw0 Fa0/3		192.168.7.66	255.255.255.192	192.168.7.126
PC1	Fa0 - Sw1 Fa0/2	LAN B	192.168.7.33	255.255.255.224	192.168.7.62

**Table 3.4 continued from previous page**

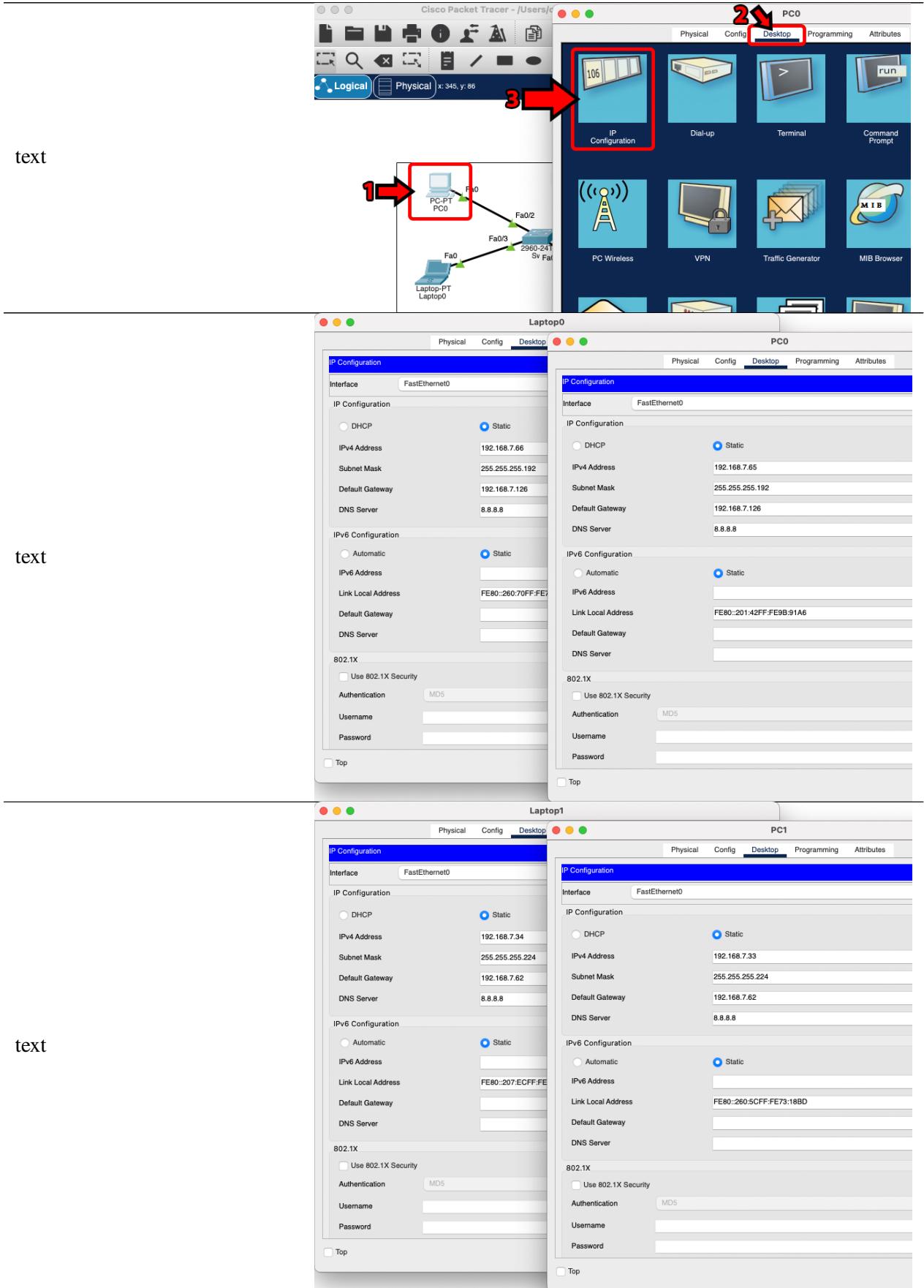
Name	Ports Link	Network	IP	Subnet Mask	Gateway
<b>Laptop1</b>	Fa0 - Sw1 Fa0/3		192.168.7.34	255.255.255.224	192.168.7.62
<b>R0</b>	Fa5/0 - R1 Fa5/0	LAN Transit B	192.168.7.5	255.255.255.252	
	Fa4/0 - R2 Fa4/0	LAN Transit C	192.168.7.9	255.255.255.252	
	Fa0/0	External			
<b>R1</b>	Fa4/0 - R2 Fa5/0	LAN Transit A	192.168.7.1	255.255.255.252	
	Fa5/0 - R1 Fa4/0	LAN Transit B	192.168.7.6	255.255.255.252	
	Fa0/0 - Sw0 Fa0/1	LAN A	192.168.7.126	255.255.255.192	
	Fa1/0 - Sw1 Fa0/1	LAN B	192.168.7.62	255.255.255.224	
<b>R2</b>	Fa5/0 - R1 Fa4/0	LAN Transit A	192.168.7.2	255.255.255.252	
	Fa4/0 - R0 Fa4/0	LAN Transit C	192.168.7.10	255.255.255.252	
	Fa0/0 - Sw2 Fa0/4	LAN Server	192.168.7.254	255.255.255.128	
<b>DHCP Server</b>	Fa0 - Sw2 Fa0/3		192.168.7.129	255.255.255.128	192.168.7.254
<b>DNS Server</b>	Fa0 - Sw2 Fa0/2	LAN Server	192.168.7.130	255.255.255.128	192.168.7.254
<b>HTTP Server</b>	Fa0 - Sw2 Fa0/1		192.168.7.131	255.255.255.128	192.168.7.254
<b>Sw0</b>	Fa0/1 - R1 Fa0/0				
	Fa0/2 - PC0	LAN A			
	Fa0/3 - Laptop0				
<b>Sw1</b>	Fa0/1 - R1 Fa1/0				
	Fa0/2 - PC1	LAN B			
	Fa0/3 - Laptop1				
<b>Sw2</b>	Fa0/1 - HTTP				
	Fa0/2 - DNS				
	Fa0/3 - DHCP	LAN Server			
	Fa0/4 - R2 Fa0/0				

Table 3.4: IP configuration table

The above tables will be used through out the project. Instead of planning for each phase and re-assigning the entire network, It was opted to fully outline every subnet and device for the sake of simplicity and work. However, here we'll focus on router **R1**, and networks **LAN A** and **LAN B**. For now let's just focus on the IP addresses, the meaning of values will be explained in Phase 3. And once again, **Cisco Packet Tracer** to the help:

## Steps

## Example



text

text

Continued on next page

**Table 3.5 continued from previous page**

**Steps**

**Example**

The screenshot shows a Cisco Packet Tracer interface. On the left, a network diagram displays two hosts: 'PC-PT PC0' and 'Laptop-PT Laptop0'. They are connected to a central switch labeled '2960-24T Sv Fa1'. The 'PC-PT' host has three interfaces: Fa0, Fa0/2, and Fa0/3. The 'Laptop-PT' host has one interface: Fa0. Red arrows labeled 1 and 2 point to the 'PC-PT' host and its Fa0 interface respectively. A red box highlights the 'Desktop' tab in the top navigation bar of a configuration window for 'FastEthernet0/0'. A red arrow labeled 3 points from the highlighted 'Desktop' tab to the 'Command Prompt' icon in the bottom right corner of the same window.

**text**

**R1**

Physical	Config	CLI	Attributes
<b>GLOBAL</b>			
Settings			
Algorithm Settings			
<b>ROUTING</b>			
Static			
RIP			
<b>INTERFACE</b>			
FastEthernet0/0			
FastEthernet1/0			
Serial2/0			
Serial3/0			
FastEthernet4/0			
FastEthernet5/0			

Port Status  
Bandwidth  
Duplex  
MAC Address

**R1**

Physical	Config	CLI	Attributes
<b>GLOBAL</b>			
Settings			
Algorithm Settings			
<b>ROUTING</b>			
Static			
RIP			
<b>INTERFACE</b>			
FastEthernet0/0			
FastEthernet1/0			
Serial2/0			
Serial3/0			
FastEthernet4/0			
FastEthernet5/0			

Tx Ring Limit

**Equivalent IOS Commands**

```

Router>enable
Press RETURN to get started!

Router>enable
Router>configure terminal
Enter configuration commands, one per line. End with Ctrl-Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.7.126 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#exit
Router(config)#exit

Router#show ip int brief
FastEthernet0/0 is up, line protocol is up
FastEthernet1/0 is up, line protocol is up
Serial2/0 is up, line protocol is up
Serial3/0 is up, line protocol is up
FastEthernet4/0 is up, line protocol is up
FastEthernet5/0 is up, line protocol is up

```

**Equivalent IOS Commands**

```

Router>enable
Press RETURN to get started!

Router>enable
Router>configure terminal
Enter configuration commands, one per line. End with Ctrl-Z.
Router(config)#interface FastEthernet1/0
Router(config-if)#ip address 192.168.7.126 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#exit
Router(config)#exit

Router#show ip int brief
FastEthernet0/0 is up, line protocol is up
FastEthernet1/0 is up, line protocol is up
Serial2/0 is up, line protocol is up
Serial3/0 is up, line protocol is up
FastEthernet4/0 is up, line protocol is up
FastEthernet5/0 is up, line protocol is up

```

**Cisco Packet Tracer - /Users/cisco/Desktop**

PCO

Physical Config Desktop Programming Attributes

Logical Physical x: 345, y: 86

1 → PC-PT PC0

2 → Fa0

3 → Command Prompt

Continued on next page

**Table 3.5 continued from previous page**

**Steps**

**Example**

The figure consists of three vertically stacked windows from the Cisco Packet Tracer application. Each window has a title bar with the host name (Laptop0 or PC0) and tabs for Physical, Config, Desktop, Programming, and Attributes. The Desktop tab is selected.

**Laptop0 (Top Window):**

```

Cisco Packet Tracer PC Command Line 1.0
C:\>arp -a
No ARP Entries Found
C:\>ping 192.168.7.65
Pinging 192.168.7.65 with 32 bytes of data:
Reply from 192.168.7.65: bytes=32 time<ms TTL=128
Ping statistics for 192.168.7.65:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.7.126
Pinging 192.168.7.126 with 32 bytes of data:
Reply from 192.168.7.126: bytes=32 time<ms TTL=255
Ping statistics for 192.168.7.126:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.7.34
Pinging 192.168.7.34 with 32 bytes of data:
Reply from 192.168.7.34: bytes=32 time<ms TTL=127
Ping statistics for 192.168.7.34:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>

```

**Laptop0 (Middle Window):**

```

Ping statistics for 192.168.7.62:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>tracert 192.168.7.65
Tracing route to 192.168.7.65 over a maximum of 30 hops:
  1  0 ms   0 ms   0 ms   192.168.7.65
Trace complete.
C:\>tracert 192.168.7.33
Tracing route to 192.168.7.33 over a maximum of 30 hops:
  1  0 ms   0 ms   0 ms   192.168.7.33
  2  0 ms   0 ms   0 ms   192.168.7.33
Trace complete.
C:\>tracert 192.168.7.34
Tracing route to 192.168.7.34 over a maximum of 30 hops:
  1  0 ms   0 ms   0 ms   192.168.7.34
  2  0 ms   0 ms   0 ms   192.168.7.34
Trace complete.
C:\>tracert 192.168.7.126
Tracing route to 192.168.7.126 over a maximum of 30 hops:
  1  0 ms   0 ms   0 ms   192.168.7.126
  2  0 ms   0 ms   0 ms   192.168.7.126
Trace complete.
C:\>tracert 192.168.7.62
Tracing route to 192.168.7.62 over a maximum of 30 hops:
  1  0 ms   0 ms   0 ms   192.168.7.62
Trace complete.
C:\>

```

**PC0 (Bottom Window):**

```

Cisco Packet Tracer PC Command Line 1.0
C:\>arp -a
No ARP Entries Found
C:\>ping 192.168.7.66
Pinging 192.168.7.66 with 32 bytes of data:
Reply from 192.168.7.66: bytes=32 time<ms TTL=128
Ping statistics for 192.168.7.66:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.7.126
Pinging 192.168.7.126 with 32 bytes of data:
Reply from 192.168.7.126: bytes=32 time<ms TTL=255
Ping statistics for 192.168.7.126:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.7.34
Pinging 192.168.7.34 with 32 bytes of data:
Reply from 192.168.7.34: bytes=32 time<ms TTL=127
Ping statistics for 192.168.7.34:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.7.33
Pinging 192.168.7.33 with 32 bytes of data:
Request timed out.
Reply from 192.168.7.33: bytes=32 time<ms TTL=127
Ping statistics for 192.168.7.33:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.7.62
Pinging 192.168.7.62 with 32 bytes of data:
Request timed out.
Reply from 192.168.7.62: bytes=32 time<ms TTL=127
Ping statistics for 192.168.7.62:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>

```

text

Continued on next page

**Table 3.5 continued from previous page**

**Steps**

**Example**

The figure consists of four vertically stacked screenshots of terminal windows, each with a title bar and a command prompt area.

- Laptop1:** Shows two terminal sessions. The left session runs Cisco Packet Tracer Command Line 1.0 and displays ping and traceroute results to 192.168.7.33 and 192.168.7.126. The right session runs C:\> and shows similar tracing results.
- PC1:** Shows two terminal sessions. The left session runs Cisco Packet Tracer PC Command Line 1.0 and displays ping and traceroute results to 192.168.7.33 and 192.168.7.126. The right session runs C:\> and shows similar tracing results.
- R1:** Shows two terminal sessions. The left session runs IOS Command Line Interface and displays configuration commands for interface 0/0/0 and 0/0/1, including ip address and mtu settings. The right session runs R1 and shows output of show ip int brief and show ip route commands.

**text**

text

text

Continued on next page

**Table 3.5 continued from previous page****Steps****Example**


---

Table 3.5: Cisco Packet Tracer guide

And to supplement the output images, below are the text versions (*considering this isn't a report for ants*):

```

1 Cisco Packet Tracer PC Command Line 1.0
2 C:>arp -a
3 No ARP Entries Found
4 C:>ping 192.168.7.66
5
6
7 Pinging 192.168.7.66 with 32 bytes of data:
8
9 Reply from 192.168.7.66: bytes=32 time<1ms TTL=128
10 Reply from 192.168.7.66: bytes=32 time<1ms TTL=128
11 Reply from 192.168.7.66: bytes=32 time<1ms TTL=128
12 Reply from 192.168.7.66: bytes=32 time<1ms TTL=128
13
14 Ping statistics for 192.168.7.66:
15     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
16 Approximate round trip times in milli-seconds:
17     Minimum = 0ms, Maximum = 0ms, Average = 0ms
18
19 C:>ping 192.168.7.126
20
21 Pinging 192.168.7.126 with 32 bytes of data:
22
23 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
24 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
25 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
26 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
27
28 Ping statistics for 192.168.7.126:
29     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
30 Approximate round trip times in milli-seconds:
31     Minimum = 0ms, Maximum = 0ms, Average = 0ms
32
33 C:>ping 192.168.7.33
34
35 Pinging 192.168.7.33 with 32 bytes of data:
36
37 Request timed out.
38 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
39 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
40 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
41
42 Ping statistics for 192.168.7.33:
43     Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
44 Approximate round trip times in milli-seconds:
45     Minimum = 0ms, Maximum = 0ms, Average = 0ms
46
47 C:>ping 192.168.7.34
48
49 Pinging 192.168.7.34 with 32 bytes of data:
50
51 Request timed out.
52 Reply from 192.168.7.34: bytes=32 time<1ms TTL=127
53 Reply from 192.168.7.34: bytes=32 time<1ms TTL=127
54 Reply from 192.168.7.34: bytes=32 time<1ms TTL=127
55
56 Ping statistics for 192.168.7.34:
57     Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
58 Approximate round trip times in milli-seconds:
```

```

59     Minimum = 0ms, Maximum = 0ms, Average = 0ms
60
61 C:\>ping 192.168.7.62
62
63 Pinging 192.168.7.62 with 32 bytes of data:
64
65 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
66 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
67 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
68 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
69
70 Ping statistics for 192.168.7.62:
71     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
72 Approximate round trip times in milli-seconds:
73     Minimum = 0ms, Maximum = 0ms, Average = 0ms
74
75 C:\>arp -a
76             Internet Address          Physical Address      Type
77             192.168.7.66            0060.7078.da66    dynamic
78             192.168.7.126          0060.2fcd.90bd    dynamic
79
80 C:\>tracert 192.168.7.66
81
82 Tracing route to 192.168.7.66 over a maximum of 30 hops:
83
84     1     18 ms      0 ms      0 ms      192.168.7.66
85
86 Trace complete.
87
88 C:\>tracert 192.168.7.126
89
90 Tracing route to 192.168.7.126 over a maximum of 30 hops:
91
92     1     0 ms      0 ms      0 ms      192.168.7.126
93
94 Trace complete.
95
96 C:\>tracert 192.168.7.33
97
98 Tracing route to 192.168.7.33 over a maximum of 30 hops:
99
100    1     0 ms      0 ms      0 ms      192.168.7.126
101    2     9 ms      0 ms      0 ms      192.168.7.33
102
103 Trace complete.
104
105 C:\>tracert 192.168.7.34
106
107 Tracing route to 192.168.7.34 over a maximum of 30 hops:
108
109    1     0 ms      0 ms      0 ms      192.168.7.126
110    2     0 ms      0 ms      0 ms      192.168.7.34
111
112 Trace complete.
113
114 C:\>tracert 192.168.7.62
115
116 Tracing route to 192.168.7.62 over a maximum of 30 hops:
117
118     1     0 ms      0 ms      0 ms      192.168.7.62
119
120 Trace complete.
121
122 C:\>

```

Listing 3.4: PC0 CMD output

```

1
2 Cisco Packet Tracer PC Command Line 1.0
3 C:\>arp -a

```

```

4 No ARP Entries Found
5 C:\>ping 192.168.7.65
6
7 Pinging 192.168.7.65 with 32 bytes of data:
8
9 Reply from 192.168.7.65: bytes=32 time<1ms TTL=128
10 Reply from 192.168.7.65: bytes=32 time<1ms TTL=128
11 Reply from 192.168.7.65: bytes=32 time<1ms TTL=128
12 Reply from 192.168.7.65: bytes=32 time<1ms TTL=128
13
14 Ping statistics for 192.168.7.65:
15     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
16 Approximate round trip times in milli-seconds:
17     Minimum = 0ms, Maximum = 0ms, Average = 0ms
18
19 C:\>ping 192.168.7.126
20
21 Pinging 192.168.7.126 with 32 bytes of data:
22
23 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
24 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
25 Reply from 192.168.7.126: bytes=32 time=1ms TTL=255
26 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
27
28 Ping statistics for 192.168.7.126:
29     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
30 Approximate round trip times in milli-seconds:
31     Minimum = 0ms, Maximum = 1ms, Average = 0ms
32
33 C:\>ping 192.168.7.33
34
35 Pinging 192.168.7.33 with 32 bytes of data:
36
37 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
38 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
39 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
40 Reply from 192.168.7.33: bytes=32 time<1ms TTL=127
41
42 Ping statistics for 192.168.7.33:
43     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
44 Approximate round trip times in milli-seconds:
45     Minimum = 0ms, Maximum = 0ms, Average = 0ms
46
47 C:\>ping 192.168.7.34
48
49 Pinging 192.168.7.34 with 32 bytes of data:
50
51 Reply from 192.168.7.34: bytes=32 time=23ms TTL=127
52 Reply from 192.168.7.34: bytes=32 time<1ms TTL=127
53 Reply from 192.168.7.34: bytes=32 time<1ms TTL=127
54 Reply from 192.168.7.34: bytes=32 time<1ms TTL=127
55
56 Ping statistics for 192.168.7.34:
57     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
58 Approximate round trip times in milli-seconds:
59     Minimum = 0ms, Maximum = 23ms, Average = 5ms
60
61 C:\>ping 192.168.7.62
62
63 Pinging 192.168.7.62 with 32 bytes of data:
64
65 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
66 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
67 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
68 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
69
70 Ping statistics for 192.168.7.62:
71     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
72 Approximate round trip times in milli-seconds:
73     Minimum = 0ms, Maximum = 0ms, Average = 0ms
74
75 C:\>arp -a
76   Internet Address          Physical Address          Type
```

```

77 192.168.7.65          0001.429b.91a6      dynamic
78 192.168.7.126         0060.2fcd.90bd      dynamic
79
80 C:\>tracert 192.168.7.65
81
82 Tracing route to 192.168.7.65 over a maximum of 30 hops:
83
84     1    0 ms      0 ms      0 ms      192.168.7.65
85
86 Trace complete.
87
88 C:\>tracert 192.168.7.33
89
90 Tracing route to 192.168.7.33 over a maximum of 30 hops:
91
92     1    0 ms      0 ms      0 ms      192.168.7.126
93     2    0 ms      0 ms      0 ms      192.168.7.33
94
95 Trace complete.
96
97 C:\>tracert 192.168.7.34
98
99 Tracing route to 192.168.7.34 over a maximum of 30 hops:
100
101    1    0 ms      0 ms      19 ms      192.168.7.126
102    2    0 ms      0 ms      0 ms      192.168.7.34
103
104 Trace complete.
105
106 C:\>tracert 192.168.7.126
107
108 Tracing route to 192.168.7.126 over a maximum of 30 hops:
109
110    1    0 ms      0 ms      0 ms      192.168.7.126
111
112 Trace complete.
113
114 C:\>tracert 192.168.7.62
115
116 Tracing route to 192.168.7.62 over a maximum of 30 hops:
117
118    1    0 ms      0 ms      0 ms      192.168.7.62
119
120 Trace complete.
121
122 C:\>

```

Listing 3.5: Laptop0 CMD output

---

```

1
2 Cisco Packet Tracer PC Command Line 1.0
3 C:\>arp -a
4 No ARP Entries Found
5 C:\>ping 192.168.7.65
6
7 Pinging 192.168.7.65 with 32 bytes of data:
8
9 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
10 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
11 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
12 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
13
14 Ping statistics for 192.168.7.65:
15     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
16 Approximate round trip times in milli-seconds:
17     Minimum = 0ms, Maximum = 0ms, Average = 0ms
18
19 C:\>ping 192.168.7.66
20
21 Pinging 192.168.7.66 with 32 bytes of data:

```

```

22
23 Reply from 192.168.7.66: bytes=32 time=1ms TTL=127
24 Reply from 192.168.7.66: bytes=32 time<1ms TTL=127
25 Reply from 192.168.7.66: bytes=32 time=1ms TTL=127
26 Reply from 192.168.7.66: bytes=32 time<1ms TTL=127
27
28 Ping statistics for 192.168.7.66:
29   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
30   Approximate round trip times in milli-seconds:
31     Minimum = 0ms, Maximum = 1ms, Average = 0ms
32
33 C:\>ping 192.168.7.34
34
35 Pinging 192.168.7.34 with 32 bytes of data:
36
37 Reply from 192.168.7.34: bytes=32 time<1ms TTL=128
38 Reply from 192.168.7.34: bytes=32 time=13ms TTL=128
39 Reply from 192.168.7.34: bytes=32 time<1ms TTL=128
40 Reply from 192.168.7.34: bytes=32 time<1ms TTL=128
41
42 Ping statistics for 192.168.7.34:
43   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
44   Approximate round trip times in milli-seconds:
45     Minimum = 0ms, Maximum = 13ms, Average = 3ms
46
47 C:\>ping 192.168.7.126
48
49 Pinging 192.168.7.126 with 32 bytes of data:
50
51 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
52 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
53 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
54 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
55
56 Ping statistics for 192.168.7.126:
57   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
58   Approximate round trip times in milli-seconds:
59     Minimum = 0ms, Maximum = 0ms, Average = 0ms
60
61 C:\>ping 192.168.7.62
62
63 Pinging 192.168.7.62 with 32 bytes of data:
64
65 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
66 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
67 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
68 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
69
70 Ping statistics for 192.168.7.62:
71   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
72   Approximate round trip times in milli-seconds:
73     Minimum = 0ms, Maximum = 0ms, Average = 0ms
74
75 C:\>arp -a
76   Internet Address        Physical Address        Type
77   192.168.7.34           0007.ec04.c4a4        dynamic
78   192.168.7.62           0090.0c00.9582        dynamic
79
80 C:\>tracert 192.168.7.65
81
82 Tracing route to 192.168.7.65 over a maximum of 30 hops:
83
84   1  0 ms        20 ms        0 ms        192.168.7.62
85   2  0 ms        0 ms        0 ms        192.168.7.65
86
87 Trace complete.
88
89 C:\>tracert 192.168.7.66
90
91 Tracing route to 192.168.7.66 over a maximum of 30 hops:
92
93   1  0 ms        0 ms        0 ms        192.168.7.62
94   2  0 ms        0 ms        0 ms        192.168.7.66

```

```

95
96 Trace complete.
97
98 C:\>tracert 192.168.7.34
99
100 Tracing route to 192.168.7.34 over a maximum of 30 hops:
101
102     1    0 ms      0 ms      0 ms      192.168.7.34
103
104 Trace complete.
105
106 C:\>tracert 192.168.7.126
107
108 Tracing route to 192.168.7.126 over a maximum of 30 hops:
109
110     1    0 ms      0 ms      0 ms      192.168.7.126
111
112 Trace complete.
113
114 C:\>tracert 192.168.7.62
115
116 Tracing route to 192.168.7.62 over a maximum of 30 hops:
117
118     1    0 ms      0 ms      0 ms      192.168.7.62
119
120 Trace complete.
121
122 C:\>

```

Listing 3.6: PC1 CMD output

```

1
2 Cisco Packet Tracer PC Command Line 1.0
3 C:\>arp -a
4   Internet Address      Physical Address      Type
5   192.168.7.33          0060.5c73.18bd      dynamic
6   192.168.7.62          0090.0c00.9582      dynamic
7
8 C:\>ping 192.168.7.65
9
10 Pinging 192.168.7.65 with 32 bytes of data:
11
12 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
13 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
14 Reply from 192.168.7.65: bytes=32 time<1ms TTL=127
15 Reply from 192.168.7.65: bytes=32 time=16ms TTL=127
16
17 Ping statistics for 192.168.7.65:
18   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
19   Approximate round trip times in milli-seconds:
20     Minimum = 0ms, Maximum = 16ms, Average = 4ms
21
22 C:\>ping 192.168.7.66
23
24 Pinging 192.168.7.66 with 32 bytes of data:
25
26 Reply from 192.168.7.66: bytes=32 time<1ms TTL=127
27 Reply from 192.168.7.66: bytes=32 time<1ms TTL=127
28 Reply from 192.168.7.66: bytes=32 time<1ms TTL=127
29 Reply from 192.168.7.66: bytes=32 time<1ms TTL=127
30
31 Ping statistics for 192.168.7.66:
32   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
33   Approximate round trip times in milli-seconds:
34     Minimum = 0ms, Maximum = 0ms, Average = 0ms
35
36 C:\>ping 192.168.7.33
37
38 Pinging 192.168.7.33 with 32 bytes of data:
39

```

```

40 Reply from 192.168.7.33: bytes=32 time<1ms TTL=128
41 Reply from 192.168.7.33: bytes=32 time<1ms TTL=128
42 Reply from 192.168.7.33: bytes=32 time<1ms TTL=128
43 Reply from 192.168.7.33: bytes=32 time<1ms TTL=128
44
45 Ping statistics for 192.168.7.33:
46     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
47 Approximate round trip times in milli-seconds:
48     Minimum = 0ms, Maximum = 0ms, Average = 0ms
49
50 C:\>arp -a
51     Internet Address          Physical Address          Type
52         192.168.7.33           0060.5c73.18bd        dynamic
53         192.168.7.62           0090.0c00.9582        dynamic
54
55 C:\>tracert 192.168.7.65
56
57 Tracing route to 192.168.7.65 over a maximum of 30 hops:
58
59     1    0 ms      0 ms      0 ms      192.168.7.62
60     2    0 ms      0 ms      0 ms      192.168.7.65
61
62 Trace complete.
63
64 C:\>tracert 192.168.7.66
65
66 Tracing route to 192.168.7.66 over a maximum of 30 hops:
67
68     1    0 ms      0 ms      0 ms      192.168.7.62
69     2    0 ms      0 ms      0 ms      192.168.7.66
70
71 Trace complete.
72
73 C:\>tracert 192.168.7.33
74
75 Tracing route to 192.168.7.33 over a maximum of 30 hops:
76
77     1    0 ms      0 ms      0 ms      192.168.7.33
78
79 Trace complete.
80
81 C:\>tracert 192.168.7.126
82
83 Tracing route to 192.168.7.126 over a maximum of 30 hops:
84
85     1    0 ms      0 ms      0 ms      192.168.7.126
86
87 Trace complete.
88
89 C:\>tracert 192.168.7.62
90
91 Tracing route to 192.168.7.62 over a maximum of 30 hops:
92
93     1    0 ms      0 ms      0 ms      192.168.7.62
94
95 Trace complete.
96
97 C:\>ping 192.168.7.126
98
99 Pinging 192.168.7.126 with 32 bytes of data:
100
101 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
102 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
103 Reply from 192.168.7.126: bytes=32 time<1ms TTL=255
104 Reply from 192.168.7.126: bytes=32 time=14ms TTL=255
105
106 Ping statistics for 192.168.7.126:
107     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
108 Approximate round trip times in milli-seconds:
109     Minimum = 0ms, Maximum = 14ms, Average = 3ms
110
111 C:\>ping 192.168.7.62
112

```

```

113 Pinging 192.168.7.62 with 32 bytes of data:
114 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
115 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
116 Reply from 192.168.7.62: bytes=32 time=15ms TTL=255
117 Reply from 192.168.7.62: bytes=32 time<1ms TTL=255
118
119
120 Ping statistics for 192.168.7.62:
121     Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
122 Approximate round trip times in milli-seconds:
123     Minimum = 0ms, Maximum = 15ms, Average = 3ms
124
125 C:>

```

Listing 3.7: Laptop1 CMD output

```

1 System Bootstrap, Version 12.1(3r)T2, RELEASE SOFTWARE (fc1)
2 Copyright (c) 2000 by cisco Systems, Inc.
3 PT 1001 (PTSC2005) processor (revision 0x200) with 60416K/5120K bytes of memory
4
5 Readonly ROMMON initialized
6
7 Self decompressing the image :
8 ##### [OK]
9
10          Restricted Rights Legend
11
12 Use, duplication, or disclosure by the Government is
13 subject to restrictions as set forth in subparagraph
14 (c) of the Commercial Computer Software - Restricted
15 Rights clause at FAR sec. 52.227-19 and subparagraph
16 (c) (1) (ii) of the Rights in Technical Data and Computer
17 Software clause at DFARS sec. 252.227-7013.
18
19          cisco Systems, Inc.
20          170 West Tasman Drive
21          San Jose, California 95134-1706
22
23
24
25 Cisco Internetwork Operating System Software
26 IOS (tm) PT1000 Software (PT1000-I-M), Version 12.2(28), RELEASE SOFTWARE (fc5)
27 Technical Support: http://www.cisco.com/techsupport
28 Copyright (c) 1986-2005 by cisco Systems, Inc.
29 Compiled Wed 27-Apr-04 19:01 by miwang
30
31 PT 1001 (PTSC2005) processor (revision 0x200) with 60416K/5120K bytes of memory
32 .
33 Processor board ID PT0123 (0123)
34 PT2005 processor: part number 0, mask 01
35 Bridging software.
36 X.25 software, Version 3.0.0.
37 4 FastEthernet/IEEE 802.3 interface(s)
38 2 Low-speed serial(sync/async) network interface(s)
39 32K bytes of non-volatile configuration memory.
40 63488K bytes of ATA CompactFlash (Read/Write)
41
42 Press RETURN to get started!
43
44
45
46 Router>enable
47 Router#
48 Router#configure terminal
49 Enter configuration commands, one per line. End with CNTL/Z.
50 Router(config)#interface FastEthernet0/0
51 Router(config-if)#ip address 255.255.255.192
52 % Incomplete command.
53 Router(config-if)#ip address 192.168.7.126 255.255.255.192
54 Router(config-if)#no shutdown

```

```

55 Router(config-if)#
56 %LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
57
58 %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
59
60 Router(config-if)#exit
61 Router(config)#interface FastEthernet1/0
62 Router(config-if)#ip address 255.255.255.224
63 % Incomplete command.
64 Router(config-if)#ip address 192.168.7.62 255.255.255.224
65 Router(config-if)#no shutdown
66 Router(config-if)#
67 %LINK-5-CHANGED: Interface FastEthernet1/0, changed state to up
68
69 %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to up
70
71 Router>ping 192.168.7.65
72
73 Type escape sequence to abort.
74 Sending 5, 100-byte ICMP Echos to 192.168.7.65, timeout is 2 seconds:
75 !!!!!
76 Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
77
78 Router>ping 192.168.7.66
79
80 Type escape sequence to abort.
81 Sending 5, 100-byte ICMP Echos to 192.168.7.66, timeout is 2 seconds:
82 !!!!!
83 Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
84
85 Router>ping 192.168.7.33
86
87 Type escape sequence to abort.
88 Sending 5, 100-byte ICMP Echos to 192.168.7.33, timeout is 2 seconds:
89 !!!!!
90 Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
91
92 Router>ping 192.168.7.34
93
94 Type escape sequence to abort.
95 Sending 5, 100-byte ICMP Echos to 192.168.7.34, timeout is 2 seconds:
96 !!!!!
97 Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
98
99 Router>ping 192.168.7.126
100
101 Type escape sequence to abort.
102 Sending 5, 100-byte ICMP Echos to 192.168.7.126, timeout is 2 seconds:
103 !!!!!
104 Success rate is 100 percent (5/5), round-trip min/avg/max = 10/12/18 ms
105
106 Router>ping 192.168.7.62
107
108 Type escape sequence to abort.
109 Sending 5, 100-byte ICMP Echos to 192.168.7.62, timeout is 2 seconds:
110 !!!!!
111 Success rate is 100 percent (5/5), round-trip min/avg/max = 0/10/22 ms
112
113 Router>traceroute 192.168.7.65
114 Type escape sequence to abort.
115 Tracing the route to 192.168.7.65
116
117     1  192.168.7.65      0 msec      0 msec      0 msec
118 Router>traceroute 192.168.7.66
119 Type escape sequence to abort.
120 Tracing the route to 192.168.7.66
121
122     1  192.168.7.66      0 msec      0 msec      0 msec
123 Router>traceroute 192.168.7.33
124 Type escape sequence to abort.
125 Tracing the route to 192.168.7.33
126
127     1  192.168.7.33      0 msec      0 msec      0 msec

```

```
128 Router>traceroute 192.168.7.34
129 Type escape sequence to abort.
130 Tracing the route to 192.168.7.34
131
132     1  192.168.7.34      0 msec      0 msec      0 msec
133 Router>traceroute 192.168.7.126
134 Type escape sequence to abort.
135 Tracing the route to 192.168.7.126
136
137     1  192.168.7.126    11 msec      0 msec      0 msec
138 Router>traceroute 192.168.7.62
139 Type escape sequence to abort.
140 Tracing the route to 192.168.7.62
141
142     1  192.168.7.62    10 msec      4 msec      4 msec
143 Router>
```

Listing 3.8: Router 1 CLI output

# **Chapter 4**

## **Issues and fixes**

Running python code:

Python3 wasn't installed by default. Then had to run the code with: \$ python3 httpsocketv3.py.

Encrypted html body in wireshark:

Initially I had to run wireshark in a remote virtual private network (VPN) connection. Fortunately I could see the VPN doing it's magic but also couldn't see the HTTP body, since it was encrypted.

Default HTTP protocol, HTTPS:

To guarantee the HTTP connection I had to disable SSLEngine in Apache2 WebServer.

# **Chapter 5**

## **Conclusions**

During phase 1 many challenges were met. By creating (or in this case adapting) a webclient without using the http library, it allowed a better understanding of the protocol requests and replies by taking advantage of the provided protocol stack in a operating system. Employing the wireless packet monitor, wireshark, concepts related to http were better understood as all transactions between webclient and webserver were seen in real time, allowing a greater furthering of knowledge.

## **Appendix A**

## **Appendix**