

# Assignment 2 - Network Analysis

Adeeb Hadi Seeyad (B190737CS)

---

## Introduction

This assignment uses a packet sniffer, Wireshark. Wireshark is a free open source network protocol analyzer. It is used for network troubleshooting and communication protocol analysis. Wireshark captures network packets in real time and displays them in human-readable format. It provides many advanced features including live capture and offline analysis, three-pane packet browser, coloring rules for analysis.

## Answers

### Answer 1

#### 1. Get the IP address

First it fetches an ip address corresponding to the domain name 'minerva.nitc.ac.in' from the default DNS server.

	IP Address
Client (Host machine)	196.168.0.112
Server (Minera NITC)	103.160.223.7

#### 2. Establish the connection

Establishes a connection to the minerva.nitc.ac.in server with a three way handshake ie,

- I. The client sends a segment with SYN bit set and a random sequence number.
-

- II. The server returns a segment with ACK and SYN bits set and a new random number as sequence number and acknowledgement number as the number previously sent by the client plus an offset of 1.
- III. The client sends a segment with ACK bit set to acknowledge the server's packet.

Wireshark packet capture showing network traffic. The filter is 'ip.addr == 103.160.223.7 || dns contains nitc'. The capture shows a series of DNS queries and responses, followed by a TCP handshake and a TLS client hello message.

No.	Time	Source	Destination	Protocol	Length	Info
65	27.515894590	192.168.0.112	192.168.0.1	DNS	78	Standard query 0x8592 A minerva.nitc.ac.in
66	27.515913068	192.168.0.112	192.168.0.1	DNS	78	Standard query 0x5391 AAAA minerva.nitc.ac.in
67	27.518018465	192.168.0.1	192.168.0.112	DNS	94	Standard query response 0x8592 A minerva.nitc.ac.in A 103.160.223.7
68	27.519683054	192.168.0.1	192.168.0.112	DNS	123	Standard query response 0x5391 AAAA minerva.nitc.ac.in SOA ebox.nitc.ac.in
69	27.519779694	192.168.0.112	103.160.223.7	TCP	74	39444 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3800957627 TSecr=...
70	27.955531430	103.160.223.7	192.168.0.112	TCP	74	443 → 39444 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1452 SACK_PERM=1 TSval=3139...
71	27.955596009	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3800958063 TSecr=3139695127
72	27.958030338	192.168.0.112	103.160.223.7	TLSv1.2	583	Client Hello

### 3. Negotiate secure exchange of data

The client and the server use a hand-shake protocol to negotiate on how to securely exchange data over the internet.

- I. Client sends a **client hello** message that lists information such as TLS version, the cryptographic algorithms and the data compression methods supported by the client machine.
- II. Server responds with a **server hello** message that chooses TLS version 1.2, a cryptographic algorithm from the list sent by the client, session ID etc.
- III. Server sends its public key to the client in a process known as **server key exchange** and the server hello is completed.
- IV. Client sends its private key to the server encrypted by server's public key in a process known as **client key exchange**.

Wireshark packet capture showing network traffic. The filter is 'ip.addr == 103.160.223.7 || dns contains nitc'. The capture shows the continuation of the TLS handshake, including server hello, server key exchange, client key exchange, and change cipher specification messages.

No.	Time	Source	Destination	Protocol	Length	Info
72	27.958030338	192.168.0.112	103.160.223.7	TLSv1.2	583	Client Hello
73	28.365171776	103.160.223.7	192.168.0.112	TCP	66	443 → 39444 [ACK] Seq=1 Ack=518 Win=64768 Len=0 TSval=3139695560 TSecr=3800958066
74	28.365173022	103.160.223.7	192.168.0.112	TLSv1.2	4162	Server Hello
75	28.365273339	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [ACK] Seq=518 Ack=4097 Win=60160 Len=0 TSval=3800958472 TSecr=31396955...
76	28.365173165	103.160.223.7	192.168.0.112	TLSv1.2	438	Certificate, Server Key Exchange, Server Hello Done
77	28.365358930	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [ACK] Seq=518 Ack=4469 Win=59904 Len=0 TSval=3800958473 TSecr=31396955...
78	28.368262076	192.168.0.112	103.160.223.7	TLSv1.2	159	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
79	28.774374031	103.160.223.7	192.168.0.112	TLSv1.2	117	Change Cipher Spec, Encrypted Handshake Message
80	28.774433244	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [ACK] Seq=611 Ack=4520 Win=64128 Len=0 TSval=3800958882 TSecr=31396959...

## 4. Data transfer

Data is transferred from server to client using TLSv1.2 protocol in the application layer and once the packet is received by the client it sends the server a TCP segment with ACK bit set to acknowledge that the right packet is received without any error. The sequence numbers and acknowledgement numbers are updated according to the data being transmitted.

## 5. Connection termination

When the client has finished downloading the PDF it initiates a connection termination for a graceful connection release.

- I. The client sends the server a segment with FIN bit set which implies the client doesn't have any more data to send to the server.
- II. The server sends back a segment with FIN and ACK bits set to acknowledge the client's segment and to close the connection.
- III. The client sends a segment to the server with ACK bit set to acknowledge the server's segment and the connection is closed.

The image shows a Wireshark packet capture window titled 'asg2-q1.pcapng'. The filter is 'ip.addr == 103.160.223.7 || dns contains nitc'. The packet list shows several TCP segments. The packet details pane is expanded for packet 113, showing the following structure:

- Frame 71: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface wlo1, id 0
- Ethernet II, Src: IntelCor\_f1:3b:23 (a0:a4:c5:f1:3b:23), Dst: D-LinkIn\_72:55:ac (78:98:e8:72:55:ac)
- Internet Protocol Version 4, Src: 192.168.0.112, Dst: 103.160.223.7
- Transmission Control Protocol, Src Port: 39444, Dst Port: 443, Seq: 1, Ack: 1, Len: 0

No.	Time	Source	Destination	Protocol	Length	Info
106	30.414011944	192.168.0.112	103.160.223.7	TCP	78	[TCP Window Update] 39444 → 443 [ACK] Seq=842 Ack=95240 Win=199552 Len=0 TSval=380...
107	30.414053636	103.160.223.7	192.168.0.112	TLSv1.2	9849	Ignored Unknown Record
108	30.414063936	192.168.0.112	103.160.223.7	TCP	78	[TCP Window Update] 39444 → 443 [ACK] Seq=842 Ack=95240 Win=210112 Len=0 TSval=380...
109	30.822909364	103.160.223.7	192.168.0.112	TCP	10146	[TCP Out-Of-Order] 443 → 39444 [ACK] Seq=95240 Ack=842 Win=64640 Len=10080 TSval=3...
110	30.822980568	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [ACK] Seq=842 Ack=143903 Win=203392 Len=0 TSval=3800960930 TSecr=31396...
111	30.827862097	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [FIN, ACK] Seq=842 Ack=143903 Win=219136 Len=0 TSval=3800960935 TSecr=...
112	31.232196959	103.160.223.7	192.168.0.112	TCP	66	443 → 39444 [FIN, ACK] Seq=143903 Ack=843 Win=64640 Len=0 TSval=3139698423 TSecr=3...
113	31.232254052	192.168.0.112	103.160.223.7	TCP	66	39444 → 443 [ACK] Seq=843 Ack=143904 Win=219136 Len=0 TSval=3800961339 TSecr=31396...

## Answer 2

### Sub answer a

IP address of source - 192.168.44.53

IP address of destination - 192.168.44.1

### Sub answer b

HTTP was the protocol used

### Sub answer c

Username - vasudevanar

Password - vasu

The image shows a Wireshark packet capture of an HTTP POST request. The packet list shows a POST request from 192.168.44.53 to 192.168.44.1. The packet details pane shows the request body is a form with the following fields:

- Form item: "4Tredir" = "http://detectportal.firefox.com/success.txt"
- Form item: "magic" = "179048ba09bf3146"
- Form item: "username" = "vasudevanar"
- Form item: "password" = "vasu"

The packet bytes pane shows the raw data of the request body, which is a URL-encoded form. The data is: `r=http%3A%2F%2Fdetectportal.firefox.com%2Fsuccess.txt&magic=179048ba09bf3146&username=vasudevanar&password=vasu`

---

## Answer 3

### Packet 27

443								59138							
3056868986															
1084580465															
5	0	0	1	0	0	0	1	60							
0x5442								0							
0															
0															

### Packet 32

59138								443							
1660956066															
3861199010															
5	0	0	1	0	1	0	0	0							
0xfaec								0							
0															
0															