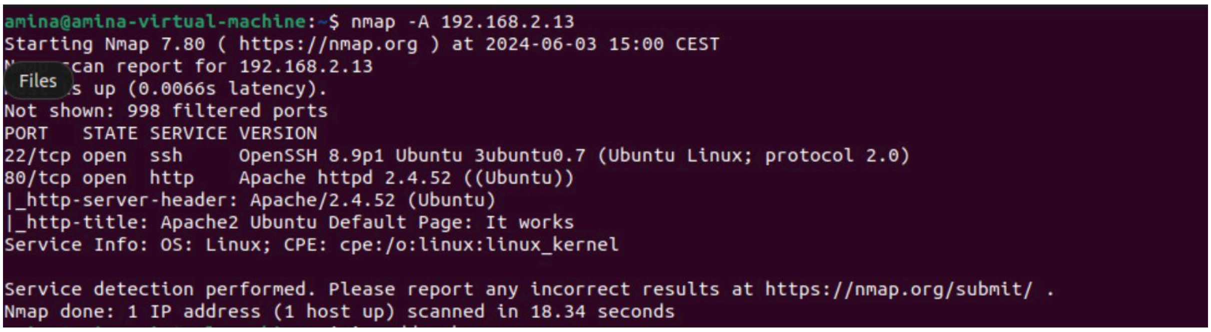
Wireshark exercise 3 Try Out

Packet Capture with Nmap:

- We started by scanning a target machine's IP address to identify open ports, services running on those ports, and any version information.
- The command used was: **nmap -A 192.168.2.13.**

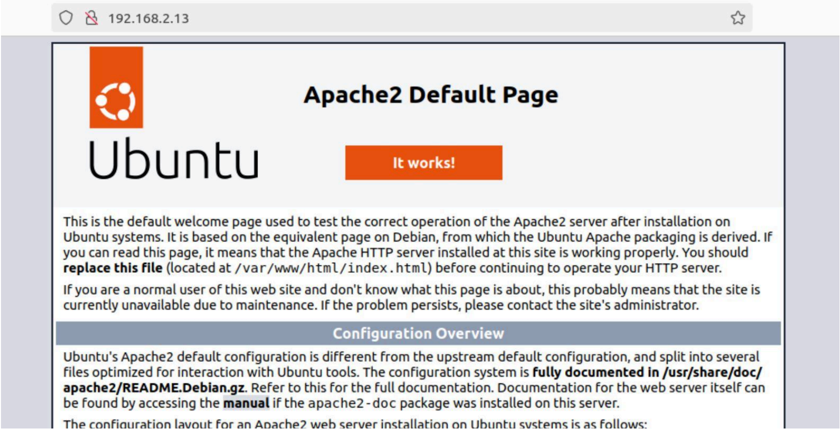


```
amina@amina-virtual-machine:~$ nmap -A 192.168.2.13
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-03 15:00 CEST
Nmap scan report for 192.168.2.13
Host is up (0.0066s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.7 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.52 ((Ubuntu))
|_http-server-header: Apache/2.4.52 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.34 seconds
```

2. Capturing Traffic with Wireshark:

- We opened Wireshark on the target machine and started capturing traffic on the network interface connected to our attacker machine.
- We generated traffic between the two machines by accessing a website running on port 80 of the target machine. I installed Apache to check if this worked and it did!:-)



Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

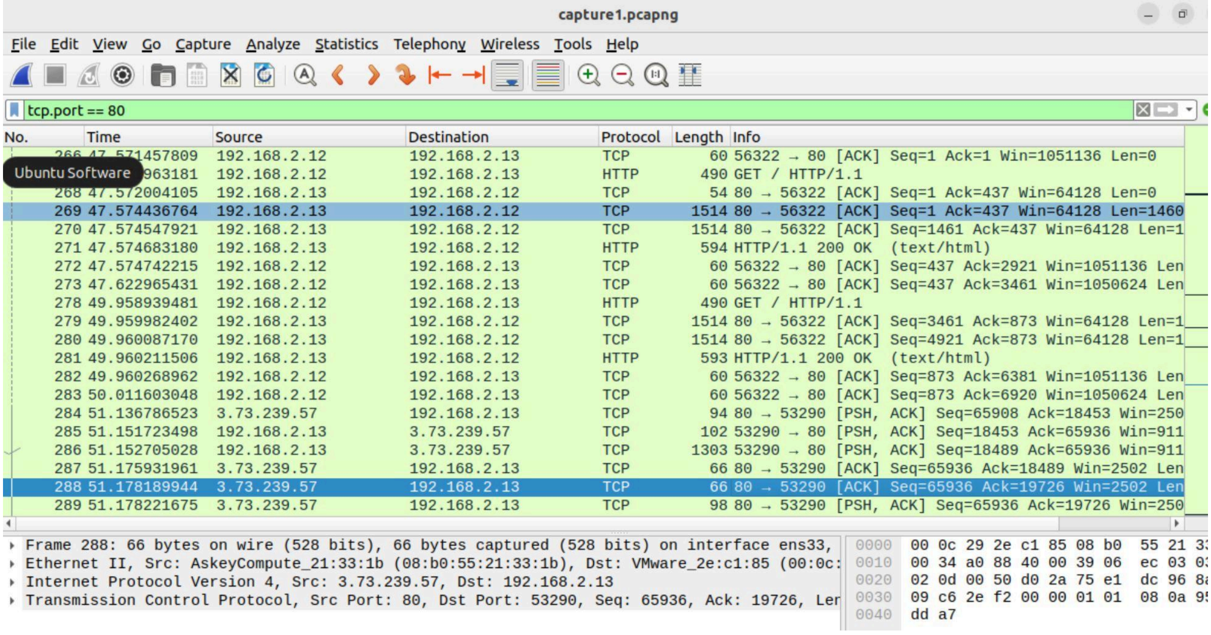
Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

3. Correlating Nmap and Wireshark Findings:

- After capturing traffic, we stopped the capture on the target machine's Wireshark and saved the captured file **(capture1.pcapng).**
- We transferred the file to our attacker machine **(LION)** using **SCP** for analysis.
- However, there were initial challenges in locating the transferred file on the attacker machine due to errors in directory paths and file naming.


capture1.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Ubuntu Software

```
tcp.port == 80
```

```
No.     Time          Source          Destination     Protocol Length Info
    266 47.571457809  192.168.2.12    192.168.2.13    TCP          60 56322 → 80 [ACK] Seq=1 Ack=1 Win=1051136 Len=0
           963181     192.168.2.12    192.168.2.13    HTTP        490 GET / HTTP/1.1
    268 47.572004105  192.168.2.13    192.168.2.12    TCP          54 80 → 56322 [ACK] Seq=1 Ack=437 Win=64128 Len=0
    269 47.574436764  192.168.2.13    192.168.2.12    TCP        1514 80 → 56322 [ACK] Seq=1 Ack=437 Win=64128 Len=1460
    270 47.574547921  192.168.2.13    192.168.2.12    TCP        1514 80 → 56322 [ACK] Seq=1461 Ack=437 Win=64128 Len=1
    271 47.574683180  192.168.2.13    192.168.2.12    HTTP        594 HTTP/1.1 200 OK  (text/html)
    272 47.574742215  192.168.2.12    192.168.2.13    TCP          60 56322 → 80 [ACK] Seq=437 Ack=2921 Win=1051136 Len
    273 47.622965431  192.168.2.12    192.168.2.13    TCP          60 56322 → 80 [ACK] Seq=437 Ack=3461 Win=1050624 Len
    278 49.958939481  192.168.2.12    192.168.2.13    HTTP        490 GET / HTTP/1.1
    279 49.959982402  192.168.2.13    192.168.2.12    TCP        1514 80 → 56322 [ACK] Seq=3461 Ack=873 Win=64128 Len=1
    280 49.960087170  192.168.2.13    192.168.2.12    TCP        1514 80 → 56322 [ACK] Seq=4921 Ack=873 Win=64128 Len=1
    281 49.960211506  192.168.2.13    192.168.2.12    HTTP        593 HTTP/1.1 200 OK  (text/html)
    282 49.960268962  192.168.2.12    192.168.2.13    TCP          60 56322 → 80 [ACK] Seq=873 Ack=6381 Win=1051136 Len
    283 50.011603048  192.168.2.12    192.168.2.13    TCP          60 56322 → 80 [ACK] Seq=873 Ack=6920 Win=1050624 Len
    284 51.136786523  3.73.239.57     192.168.2.13    TCP          94 80 → 53290 [PSH, ACK] Seq=65908 Ack=18453 Win=250
    285 51.151723498  192.168.2.13    3.73.239.57     TCP         102 53290 → 80 [PSH, ACK] Seq=18453 Ack=65936 Win=911
    286 51.152705028  192.168.2.13    3.73.239.57     TCP        1303 53290 → 80 [PSH, ACK] Seq=18489 Ack=65936 Win=911
    287 51.175931961  3.73.239.57     192.168.2.13    TCP          66 80 → 53290 [ACK] Seq=65936 Ack=18489 Win=2502 Len
    288 51.178189944  3.73.239.57     192.168.2.13    TCP          66 80 → 53290 [ACK] Seq=65936 Ack=19726 Win=2502 Len
    289 51.178221675  3.73.239.57     192.168.2.13    TCP          98 80 → 53290 [PSH, ACK] Seq=65936 Ack=19726 Win=250
```

```
▸ Frame 288: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface ens33,
▸ Ethernet II, Src: AskeyCompute_21:33:1b (08:b0:55:21:33:1b), Dst: VMware_2e:c1:85 (00:0c:
▸ Internet Protocol Version 4, Src: 3.73.239.57, Dst: 192.168.2.13
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 53290, Seq: 65936, Ack: 19726, Ler
```

```
0000  00 0c 29 2e c1 85 08 b0  55 21 33
0010  00 34 a0 88 40 00 39 06  ec 03 03
0020  02 0d 00 50 d0 2a 75 e1  dc 96 8a
0030  09 c6 2e f2 00 00 01 01  08 0a 95
0040  dd a7
```

4. Analyzing Captured Traffic:

- Once we located the file, we opened it in Wireshark on our attacker machine for analysis.
- Wireshark displayed the captured packets, allowing us to identify the communication between the attacker and target machines.
- We compared the details in the captured packets with the findings from the Nmap scan to gain insights into the network activity.

Despite encountering initial challenges with file transfer and directory paths, we successfully analyzed the captured traffic and correlated the findings from Nmap and Wireshark.