

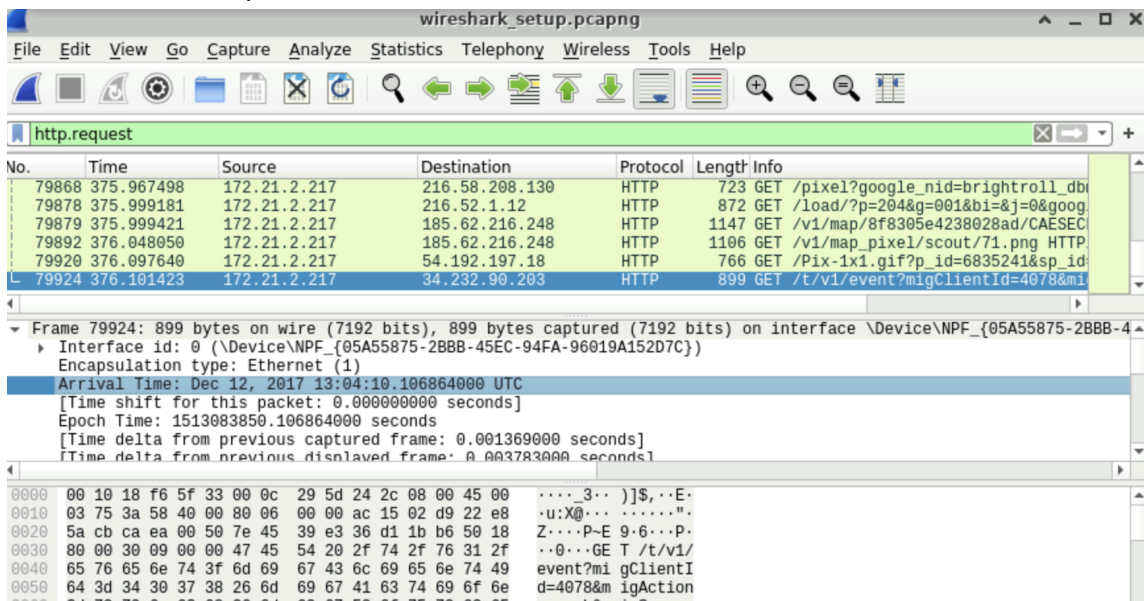
# Network Capture, Analysis, and Scanning - Lab 2 - Arghya Sarkar - abs9425



Explanation:

1. Raw data is too much info to display. With port numbers we can identify common protocols and services associated with them.
2. tcp.port eq 25 filters for SMTP traffic, which typically uses port 25
3. Filter for http.request (open pcapng file)

4. Double click on request and look at info



5. Look at the destination of request

## Congratulations Arghya

You have completed "Packet Analysis: Packet Capture Basics"



**+100**  
Total Points  
**200**

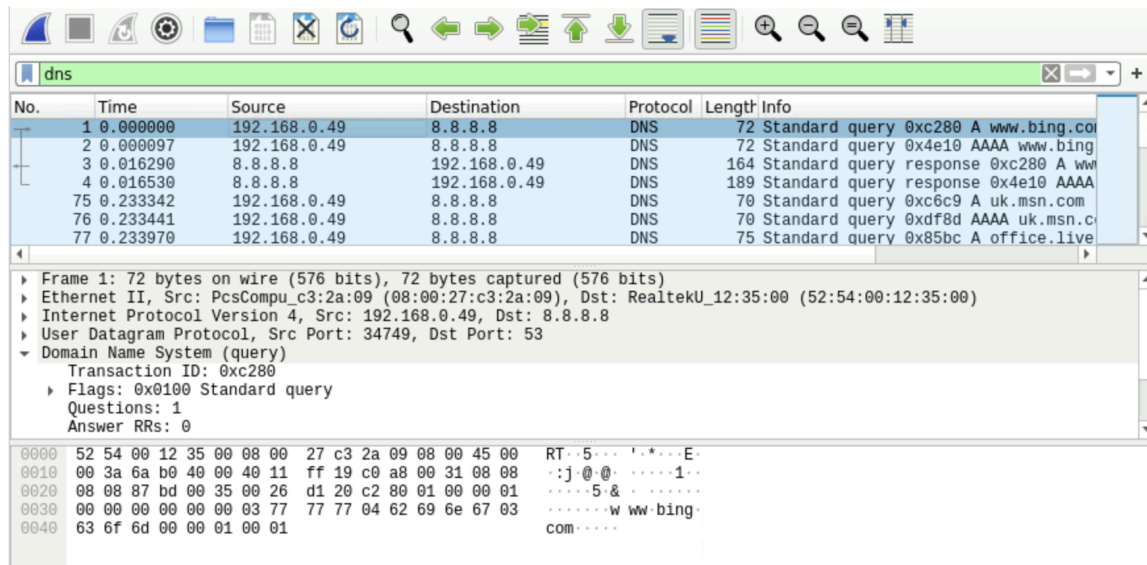


**2/3**  
You've completed 2 of your 3  
labs for this week!

**What did you think of the lab?**

Explanation:

1. Filter for "dns", select the first DNS packet, expand "Domain Name System (query)" in the packet details, and check under "Queries"



2. Open the response packet, expand the "Domain Name System (query)" section, and locate the IP address in the query results

```

Authority RRs: 0
Additional RRs: 0
Queries
  www.bing.com: type A, class IN
  [Response in: 3]

```

3. To get user agents analyze GET requests. Filter for GET requests (http.request.method == "GET"), select the first HTTP packet, expand Hypertext Transfer Protocol in the packet details pane, and locate the user-agent field

```

User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.7.1\r\n
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n

```

4. Filter for "http.server", select a packet, expand "Hypertext Transfer Protocol" in the details pane, and locate the "Server" heading

```

Server: Microsoft-IIS/8.5\r\n

```

5. Use "Find Packet" (in Edit menu) with the PNG filename, locate the GET request, and note the response packet number. Export HTTP objects, filter for PNG, save the specific response packet, and open the downloaded image to view the required text

```

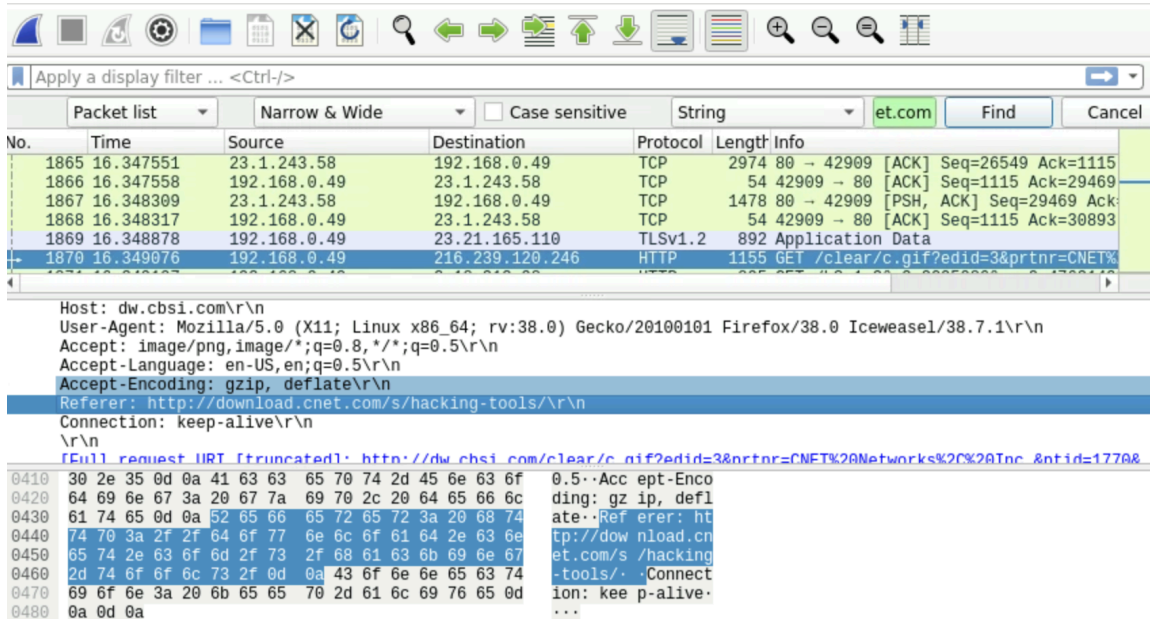
1722 dl1.cbsistatic.com image/png
1724 a3.fdlstatic.com image/png
1726 a3.fdlstatic.com image/png
1728 a3.fdlstatic.com image/png
1742 dl1.cbsistatic.com image/png
1744 dl1.cbsistatic.com image/png
1756 i.i.cbsi.com image/png
1796 i.i.cbsi.com image/png
1847 a1.fdlstatic.com image/png

```

6. To find the number of conversations, use Wireshark's Statistics > Conversations feature and check the total count in the IPv4 tab or overall display

1	IPv4 · 89	IPv6
▼	Address B	F
98	192.168.0.49	
	192.168.0.49	
110	192.168.0.49	

- To find the searched term, locate the packet containing "download.cnet.com" and check the referrer heading in its Hypertext Transfer Protocol tab



Apply a display filter ... <Ctrl-/>

Packet list   Narrow & Wide   ☐ Case sensitive   String   **et.com**   Find   Cancel

No.	Time	Source	Destination	Protocol	Length	Info
1865	16.347551	23.1.243.58	192.168.0.49	TCP	2974	80 → 42909 [ACK] Seq=26549 Ack=1115
1866	16.347558	192.168.0.49	23.1.243.58	TCP	54	42909 → 80 [ACK] Seq=1115 Ack=29469
1867	16.348309	23.1.243.58	192.168.0.49	TCP	1478	80 → 42909 [PSH, ACK] Seq=29469 Ack=
1868	16.348317	192.168.0.49	23.1.243.58	TCP	54	42909 → 80 [ACK] Seq=1115 Ack=30893
1869	16.348878	192.168.0.49	23.21.165.110	TLSv1.2	892	Application Data
1870	16.349076	192.168.0.49	216.239.120.246	HTTP	1155	GET /clear/c.gif?edid=3&prtnr=CNET%20Networks%20Inc.&ntid=1770&

Host: dw.cbsi.com\r\n  
 User-Agent: Mozilla/5.0 (X11; Linux x86\_64; rv:38.0) Gecko/20100101 Firefox/38.0 Iceweasel/38.7.1\r\n  
 Accept: image/png,image/\*;q=0.8,\*/\*;q=0.5\r\n  
 Accept-Language: en-US,en;q=0.5\r\n  
 Accept-Encoding: gzip, deflate\r\n  
 Referer: http://download.cnet.com/s/hacking-tools/\r\n  
 Connection: keep-alive\r\n  
 \r\n  
 [Full request URI [truncated]: http://dw.cbsi.com/clear/c.gif?edid=3&prtnr=CNET%20Networks%20Inc.&ntid=1770&]

Offset	Time	Source	Destination	Protocol	Length	Info
0410	30.2e.35.0d.0a.41.63.63.65.70.74.2d.45.6e.63.6f			0.5.0.0	Acc	ept-Enco
0420	64.69.6e.67.3a.20.67.7a.69.70.2c.20.64.65.66.6c			ding:	gz	ip, defl
0430	61.74.65.0d.0a.52.65.66.65.72.65.72.3a.20.68.74			ate.	Ref	erer: ht
0440	74.70.3a.2f.2f.64.6f.77.6e.6c.6f.61.64.2e.63.6e			tp://dow	nload.cn	
0450	65.74.2e.63.6f.6d.2f.73.2f.68.61.63.6b.69.6e.67			et.com/s	/hacking	
0460	2d.74.6f.6f.6c.73.2f.0d.0a.43.6f.6e.6e.65.63.74			-tools/.	.Connect	
0470	69.6f.6e.3a.20.6b.65.65.70.2d.61.6c.69.76.65.0d			ion: kee	p-alive.	
0480	0a.0d.0a			...		

# Congratulations Arghya

You have completed "Packet Analysis: Using tcpdump"



**+200**

Total Points

**400**



**3/3**

You've completed 3 of your 3  
labs for this week!

**What did you think of the lab?**

Explanation:

1. Based on the briefing: "tcpdump allows you to output your results into a specified file type such as csv or txt. To do this you can specify the -w option"
2. Run `tcpdump -D`

```
linux@tcpdump:~$ tcpdump -D
1.eth0 [Up, Running]
2.lo [Up, Running, Loopback]
3.any (Pseudo-device that captures on all interfaces) [Up, Running]
4.bluetooth-monitor (Bluetooth Linux Monitor) [none]
5.nflog (Linux netfilter log (NFLOG) interface) [none]
6.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
```

3. -X form cheatsheet

4. Run `tcpdump -r tcpdump.pcap 'host 88.221.88.59'`

```
linux@tcpdump:~$ tcpdump -r tcpdump.pcap 'host 88.221.88.59'
reading from file tcpdump.pcap, link-type EN10MB (Ethernet)
07:31:56.197987 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 2054538429, win 30016, length 0
07:31:56.198136 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
07:32:06.438054 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 1, win 30016, length 0
07:32:06.438365 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
07:32:16.677955 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 1, win 30016, length 0
07:32:16.678082 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
07:32:26.921868 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 1, win 30016, length 0
07:32:26.921990 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
07:32:37.158275 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 1, win 30016, length 0
07:32:37.158725 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
07:32:47.397977 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 1, win 30016, length 0
07:32:47.398547 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
07:32:57.638112 IP ip-192-168-21-133.eu-west-1.compute.internal.40646 > a88-221-88-59.deploy.static.akamaitechnologies.com.80: Flags [.] , ack 1, win 30016, length 0
07:32:57.638538 IP a88-221-88-59.deploy.static.akamaitechnologies.com.80 > ip-192-168-21-133.eu-west-1.compute.internal.40646: Flags [.] , ack 1, win 64240, length 0
```

5. Run

`tcpdump -r tcpdump.pcap -w filtered_packets.pcap 'host 184.107.41.72 and port 80'`  
`md5sum filtered_packets.pcap`

```
linux@tcpdump:~$ tcpdump -r tcpdump.pcap -w filtered_packets.pcap 'host 184.107.41.72 and port 80'
reading from file tcpdump.pcap, link-type EN10MB (Ethernet)
linux@tcpdump:~$ md5sum filtered_packets.pcap
8e4b92724d9034a49cf10f6b147ac482  filtered_packets.pcap
```

# Congratulations Arghya

You have completed "Wireshark: Display Filters – Diving In"



**+200**

Total Points  
**600**



**4/3**

You've completed 4 of your 3  
labs for this week!

**What did you think of the lab?**

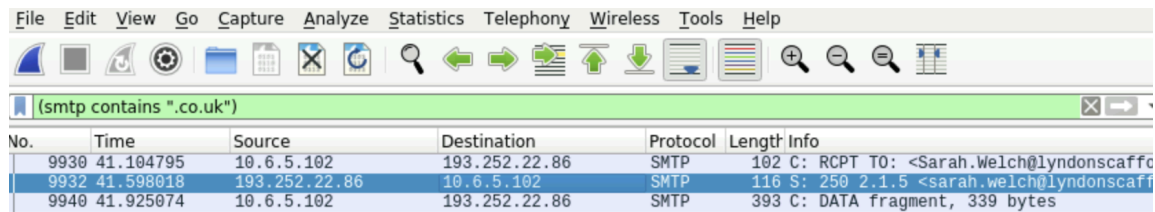
Explanation:

1. Apply the filter "(smtp contains \"Subject: \")\", open the first packet, and find the recipient's name under the \"To\" heading in the \"Simple Mail Transfer Protocol\" field

```
▶ Transmission Control Protocol, Src Port: 49269, Dst Port: 587, ...
▼ Simple Mail Transfer Protocol
  ▼ Line-based text data (23 lines)
    From: Bob Barton <ericrene.malherbe@wanadoo.fr>\r\n
    To: Sarah.Wells2@stockport.nhs.uk\r\n
    Message-ID: <2375801327.20186518318@stockport.nhs.uk>\r\n
    Subject: New Invoice / LN59175 / UR# 5010\r\n
    MIME-Version: 1.0\r\n
    Content-Type: multipart/mixed; boundary="==== NextPart 06
0000 00 5F 00 50 00 54 00 00 00 45 47 50 00 00 45 00  *
```



- Filter by "(smtp contains ".co.uk")", add response packet number, and check STMP field for packet count



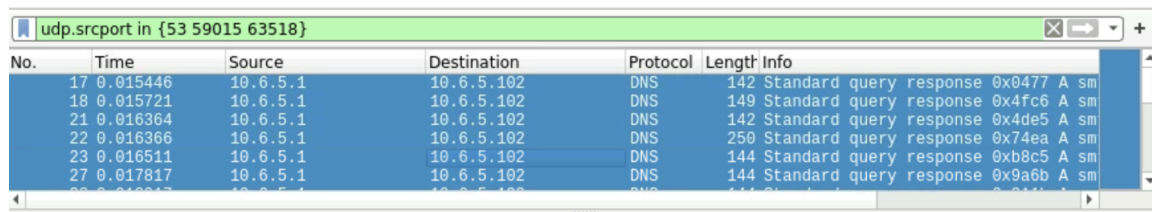
No.	Time	Source	Destination	Protocol	Length	Info
9930	41.104795	10.6.5.102	193.252.22.86	SMTP	102	C: RCPT TO: <Sarah.Welch@lyndonscaffolding.co.uk>
9932	41.598018	193.252.22.86	10.6.5.102	SMTP	116	S: 250 2.1.5 <sarah.welch@lyndonscaffolding.co.uk> recipient ok\r\n
9940	41.925074	10.6.5.102	193.252.22.86	SMTP	393	C: DATA fragment, 339 bytes

```

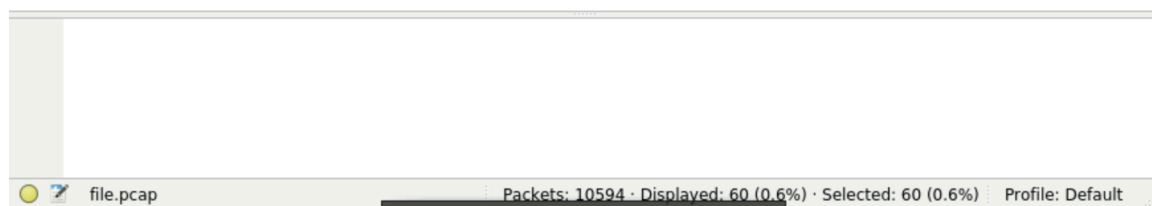
Frame 9932: 116 bytes on wire (928 bits), 116 bytes captured (928 bits) on interface 0
Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: HewlettP_1c:47:ae (00:08:02:1c:47:ae)
Internet Protocol Version 4, Src: 193.252.22.86, Dst: 10.6.5.102
Transmission Control Protocol, Src Port: 587, Dst Port: 49269, Seq: 937, Ack: 487647, Len: 62
Simple Mail Transfer Protocol
  Response: 250 2.1.5 <sarah.welch@lyndonscaffolding.co.uk> recipient ok\r\n

```

- Use `udp.srcport in {53 59015 63518}` to filter UDP source ports; both yield identical results visible in the packet count. Then use cmd + A



No.	Time	Source	Destination	Protocol	Length	Info
17	0.015446	10.6.5.1	10.6.5.102	DNS	142	Standard query response 0x0477 A sm
18	0.015721	10.6.5.1	10.6.5.102	DNS	149	Standard query response 0x4fc6 A sm
21	0.016364	10.6.5.1	10.6.5.102	DNS	142	Standard query response 0x4de5 A sm
22	0.016366	10.6.5.1	10.6.5.102	DNS	250	Standard query response 0x74ea A sm
23	0.016511	10.6.5.1	10.6.5.102	DNS	144	Standard query response 0xb8c5 A sm
27	0.017817	10.6.5.1	10.6.5.102	DNS	144	Standard query response 0x9a6b A sm



- Look at the frame index
- Look at the frame index. Default is 0



# Congratulations Arghya

You have completed "Packet Analysis: BPF Syntax"



**+100**

Total Points  
**700**



**5/3**

You've completed 5 of your 3  
labs for this week!

**What did you think of the lab?**

Explanation:

1. Abbr.
2. The two primitives in this expression are: `wlan.addr == c5:52:7e:95:6:8d` AND `wlan.fc.type\_subtype == 0x02`
3. Run `tcpdump -r bpf-pcap.pcapng 'host 10.0.50.227 and tcp port 80'`

```
west-1.compute.amazonaws.com.80: Flags [P.], seq 1:386, ack 1, win 259, length 385: HTTP: GET /5/c=10025/camp_int=Advertiser-153172%5ECampaign-814780%5Eimpressions HTTP/1.1
```

4. Run `tcpdump -r bpf-pcap.pcapng udp port 57190` and check last packet

```
11:54:43.808109 IP ip-10-0-50-227.eu-west-1.compute.internal.57190 > lhr35s07-in-f3.1e100.net.443: UDP, length 41
```

5. Run (Note DNS = 53)

`tcpdump -r bpf-pcap.pcapng -w filtered_packets.pcap not port 53 and not tcp`  
`md5sum filtered_packets.pcap`

```
linux@bpf-syntax:~$ tcpdump -r bpf-pcap.pcapng -w filtered_packets.pcap not port 53 and not tcp
reading from file bpf-pcap.pcapng, link-type EN10MB (Ethernet)
linux@bpf-syntax:~$ md5sum filtered_packets.pcap
b942d25b012745422c1719ac26419da6  filtered_packets.pcap
```

I have explained everything thoroughly and you can easily run my commands to verify results.