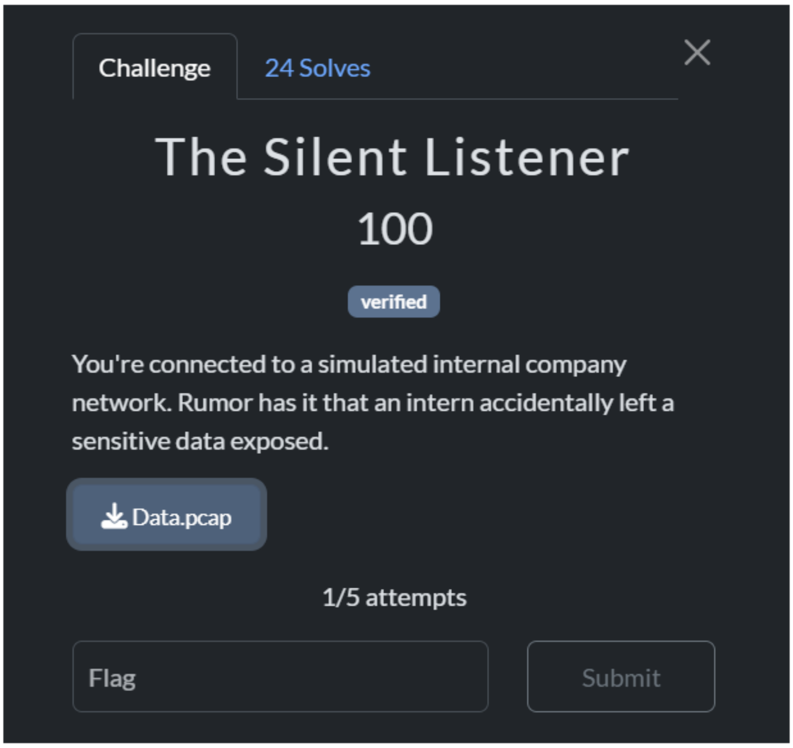
Challenge Name: The Silent Listener — 100pts (Forensics)

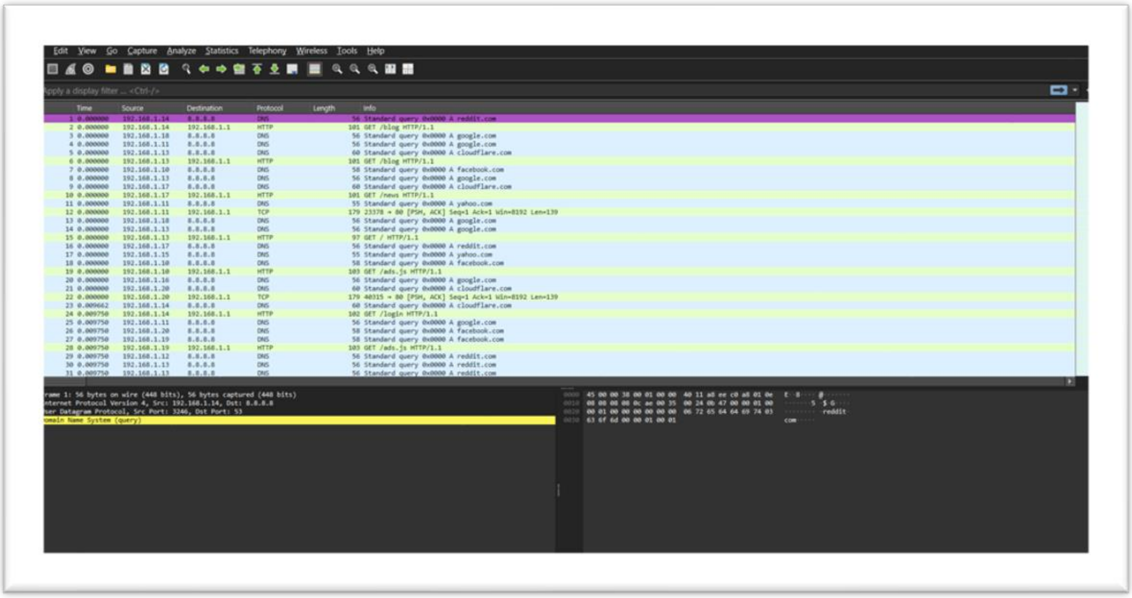


The Silent Listener

100

You're connected to a simulated internal company network. Rumor has it that an intern accidentally left a sensitive data exposed.

Here I got the pcap file, I dowmloaded that and start investigating it.



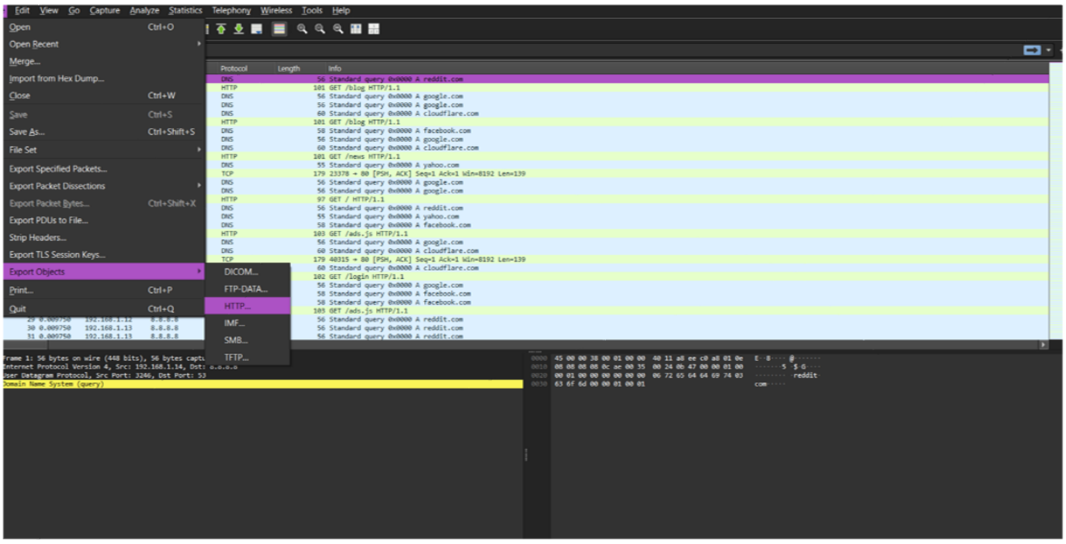
Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
pply a display filter ... <Ctrl-/>

```
     Time      Source        Destination   Protocol  Length  Info
   1 0.000000  192.168.1.14  8.8.8.8       DNS           56 Standard query 0x0000 A reddit.com
   2 0.000000  192.168.1.14  192.168.1.1   HTTP         101 GET /blog HTTP/1.1
   3 0.000000  192.168.1.18  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
   4 0.000000  192.168.1.11  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
   5 0.000000  192.168.1.13  8.8.8.8       DNS           60 Standard query 0x0000 A cloudflare.com
   6 0.000000  192.168.1.13  192.168.1.1   HTTP         101 GET /blog HTTP/1.1
   7 0.000000  192.168.1.10  8.8.8.8       DNS           58 Standard query 0x0000 A facebook.com
   8 0.000000  192.168.1.13  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
   9 0.000000  192.168.1.17  8.8.8.8       DNS           60 Standard query 0x0000 A cloudflare.com
  10 0.000000  192.168.1.17  192.168.1.1   HTTP         101 GET /news HTTP/1.1
  11 0.000000  192.168.1.11  8.8.8.8       DNS           55 Standard query 0x0000 A yahoo.com
  12 0.000000  192.168.1.11  192.168.1.1   TCP          179 23378 → 80 [PSH, ACK] Seq=1 Ack=1 Win=8192 Len=139
  13 0.000000  192.168.1.18  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
  14 0.000000  192.168.1.13  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
  15 0.000000  192.168.1.13  192.168.1.1   HTTP          97 GET / HTTP/1.1
  16 0.000000  192.168.1.17  8.8.8.8       DNS           56 Standard query 0x0000 A reddit.com
  17 0.000000  192.168.1.15  8.8.8.8       DNS           55 Standard query 0x0000 A yahoo.com
  18 0.000000  192.168.1.10  8.8.8.8       DNS           58 Standard query 0x0000 A facebook.com
  19 0.000000  192.168.1.10  192.168.1.1   HTTP         103 GET /ads.js HTTP/1.1
  20 0.000000  192.168.1.16  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
  21 0.000000  192.168.1.20  8.8.8.8       DNS           60 Standard query 0x0000 A cloudflare.com
  22 0.000000  192.168.1.20  192.168.1.1   TCP          179 40315 → 80 [PSH, ACK] Seq=1 Ack=1 Win=8192 Len=139
  23 0.009662  192.168.1.14  8.8.8.8       DNS           60 Standard query 0x0000 A cloudflare.com
  24 0.009750  192.168.1.14  192.168.1.1   HTTP         102 GET /login HTTP/1.1
  25 0.009750  192.168.1.11  8.8.8.8       DNS           56 Standard query 0x0000 A google.com
  26 0.009750  192.168.1.20  8.8.8.8       DNS           58 Standard query 0x0000 A facebook.com
  27 0.009750  192.168.1.19  8.8.8.8       DNS           58 Standard query 0x0000 A facebook.com
  28 0.009750  192.168.1.19  192.168.1.1   HTTP         103 GET /ads.js HTTP/1.1
  29 0.009750  192.168.1.12  8.8.8.8       DNS           56 Standard query 0x0000 A reddit.com
  30 0.009750  192.168.1.13  8.8.8.8       DNS           56 Standard query 0x0000 A reddit.com
  31 0.009750  192.168.1.13  8.8.8.8       DNS           56 Standard query 0x0000 A reddit.com
```

```
rame 1: 56 bytes on wire (448 bits), 56 bytes captured (448 bits)
nternet Protocol Version 4, Src: 192.168.1.14, Dst: 8.8.8.8
ser Datagram Protocol, Src Port: 3246, Dst Port: 53
omain Name System (query)
```

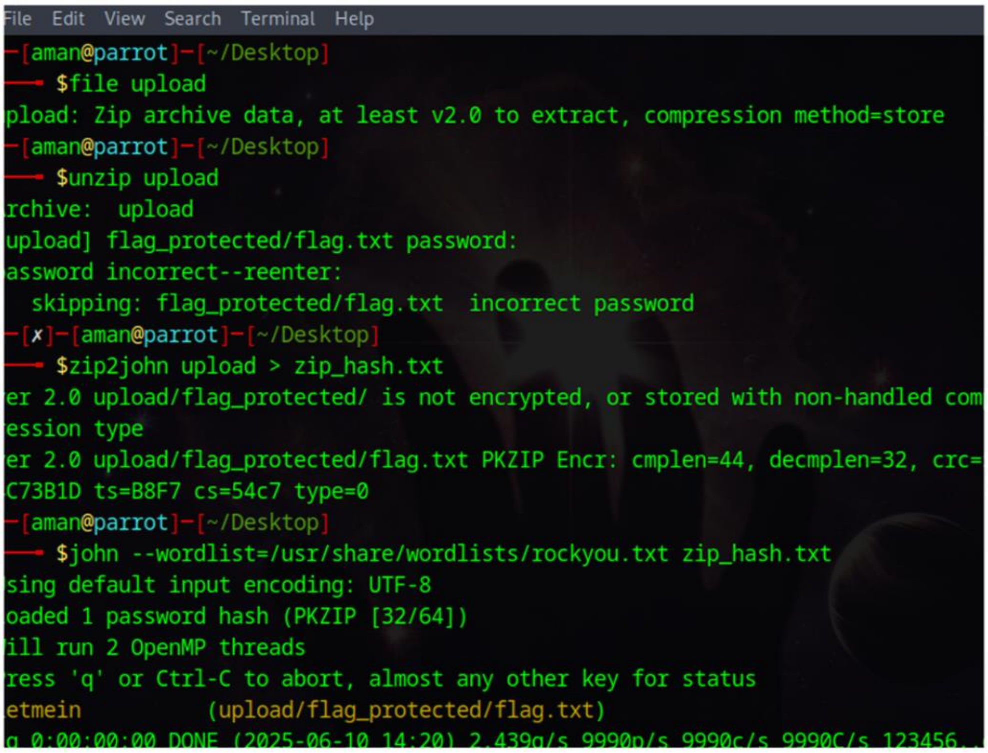
```
0000  45 00 00 38 00 01 00 00  40 11 a8 ee c0 a8 01 0e   E··8····@·······
0010  08 08 08 08 0c ae 00 35  00 24 0b 47 00 00 01 00   ·······5·$·G····
0020  00 01 00 00 00 00 00 00  06 72 65 64 64 69 74 03   ·········reddit·
0030  63 6f 6d 00 00 01 00 01                            com·····
```

My observation sees its as a most of the used protocols are http so I start to check for the http objects to export that.



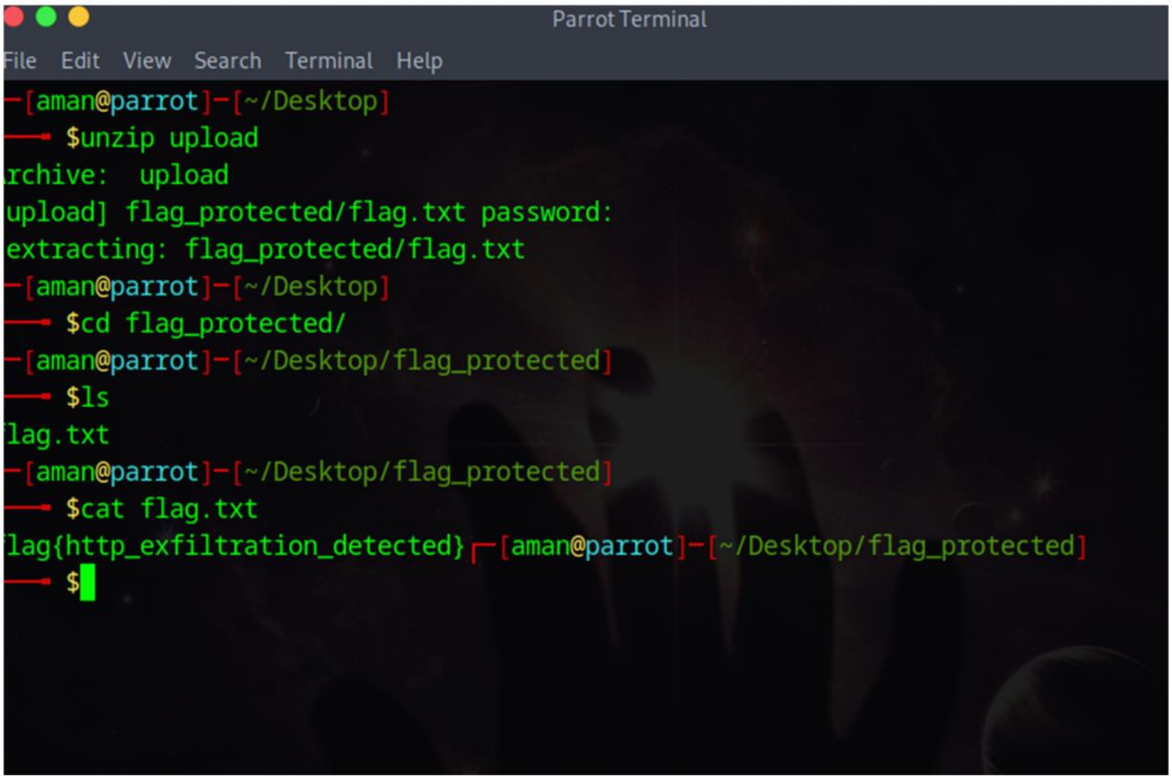
```
DNS      56 Standard query 0x0000 A reddit.com
HTTP    101 GET /blog HTTP/1.1
DNS      56 Standard query 0x0000 A google.com
DNS      56 Standard query 0x0000 A google.com
DNS      60 Standard query 0x0000 A cloudflare.com
HTTP    101 GET /blog HTTP/1.1
DNS      58 Standard query 0x0000 A facebook.com
DNS      56 Standard query 0x0000 A google.com
DNS      60 Standard query 0x0000 A cloudflare.com
HTTP    101 GET /news HTTP/1.1
DNS      55 Standard query 0x0000 A yahoo.com
TCP     179 23378 → 80 [PSH, ACK] Seq=1 Ack=1 Win=8192 Len=139
DNS      56 Standard query 0x0000 A google.com
DNS      56 Standard query 0x0000 A google.com
HTTP     97 GET / HTTP/1.1
DNS      56 Standard query 0x0000 A reddit.com
DNS      55 Standard query 0x0000 A yahoo.com
DNS      58 Standard query 0x0000 A facebook.com
HTTP    103 GET /ads.js HTTP/1.1
DNS      56 Standard query 0x0000 A google.com
DNS      60 Standard query 0x0000 A cloudflare.com
TCP     179 40315 → 80 [PSH, ACK] Seq=1 Ack=1 Win=8192 Len=139
         60 Standard query 0x0000 A cloudflare.com
        102 GET /login HTTP/1.1
         56 Standard query 0x0000 A google.com
         58 Standard query 0x0000 A facebook.com
         58 Standard query 0x0000 A facebook.com
        103 GET /ads.js HTTP/1.1
29 0.009750   192.168.1.12   8.8.8.8      56 Standard query 0x0000 A reddit.com
30 0.009750   192.168.1.13   8.8.8.8      56 Standard query 0x0000 A reddit.com
31 0.009750   192.168.1.13   8.8.8.8      56 Standard query 0x0000 A reddit.com

Frame 1: 56 bytes on wire (448 bits), 56 bytes captu
Internet Protocol Version 4, Src: 192.168.1.14, Dst: 8.8.8.8
User Datagram Protocol, Src Port: 3246, Dst Port: 53
Domain Name System (query)

0000  45 00 00 38 00 01 00 00  40 11 a8 ee c0 a8 01 0e   E··8····@·······
0010  08 08 08 08 0c ae 00 35  00 24 0b 47 00 00 01 00   ·······5·$·G····
0020  00 01 00 00 00 00 00 00  06 72 65 64 64 69 74 03   ·········reddit·
0030  63 6f 6d 00 00 01 00 01                            com·····
```

When I export the http object, I got the zip archieve file which was password proctected. I tried to crack the zip password using zip2john, zip password cracker.



```
-[aman@parrot]-[~/Desktop]
 ──╼ $file upload
pload: Zip archive data, at least v2.0 to extract, compression method=store
-[aman@parrot]-[~/Desktop]
 ──╼ $unzip upload
rchive:  upload
upload] flag_protected/flag.txt password:
assword incorrect--reenter:
  skipping: flag_protected/flag.txt  incorrect password
-[✗]-[aman@parrot]-[~/Desktop]
 ──╼ $zip2john upload > zip_hash.txt
er 2.0 upload/flag_protected/ is not encrypted, or stored with non-handled com
ession type
er 2.0 upload/flag_protected/flag.txt PKZIP Encr: cmplen=44, decmplen=32, crc=
C73B1D ts=B8F7 cs=54c7 type=0
-[aman@parrot]-[~/Desktop]
 ──╼ $john --wordlist=/usr/share/wordlists/rockyou.txt zip_hash.txt
sing default input encoding: UTF-8
oaded 1 password hash (PKZIP [32/64])
ill run 2 OpenMP threads
ress 'q' or Ctrl-C to abort, almost any other key for status
etmein          (upload/flag_protected/flag.txt)
g 0:00:00:00 DONE (2025-06-10 14:20) 2.439g/s 9990p/s 9990c/s 9990C/s 123456..
```

I got the password in the Rockyou.txt by using zip2jojn as **letmein**. So now I used that password to get the zip file open and I got flag.txt inside the zip that contains the flag.



```
─[aman@parrot]─[~/Desktop]
──╼ $unzip upload
rchive:  upload
upload] flag_protected/flag.txt password:
extracting: flag_protected/flag.txt
─[aman@parrot]─[~/Desktop]
──╼ $cd flag_protected/
─[aman@parrot]─[~/Desktop/flag_protected]
──╼ $ls
lag.txt
─[aman@parrot]─[~/Desktop/flag_protected]
──╼ $cat flag.txt
lag{http_exfiltration_detected}┌─[aman@parrot]─[~/Desktop/flag_protected]
──╼ $
```

The flag.txt contains that what we want as a flag for the solution.

FLAG: {http_exfiltration_detected}

~By Team justahacker