

Forensics CTF 8

Platform: picoCTF 2019

Challenge Name: WebNet1

Category: Forensics

Difficulty: Hard

Submitted By: Gurleen Kaur Brar

Objective

The goal was to analyze encrypted traffic in a packet capture (PCAP) file and extract a hidden flag using a provided RSA private key. The challenge involved configuring Wireshark to decrypt TLS traffic and inspecting reassembled SSL packets.

Challenge Description

WebNet1

Hard Forensics picoCTF 2019

AUTHOR: JASON

Description

We found this [packet capture](#) and [key](#). Recover the flag.

Hints ?

1 2

4,357 users solved

94% Liked

picoCTF{FLAG}

Submit Flag

Files and Tools Used

- **Files Provided:**

- `capture.pcap` – a packet capture file
- `picopico.key` – RSA private key for decrypting TLS

- **Tools Used:**
 - Kali Linux
 - Wireshark
 - RSA Key configuration panel in Wireshark

Step-by-Step Process

Step 1: View the RSA Key

Verified that the provided `picopico.key` was a standard PEM-encoded private RSA key.

```
cat picopico.key
```

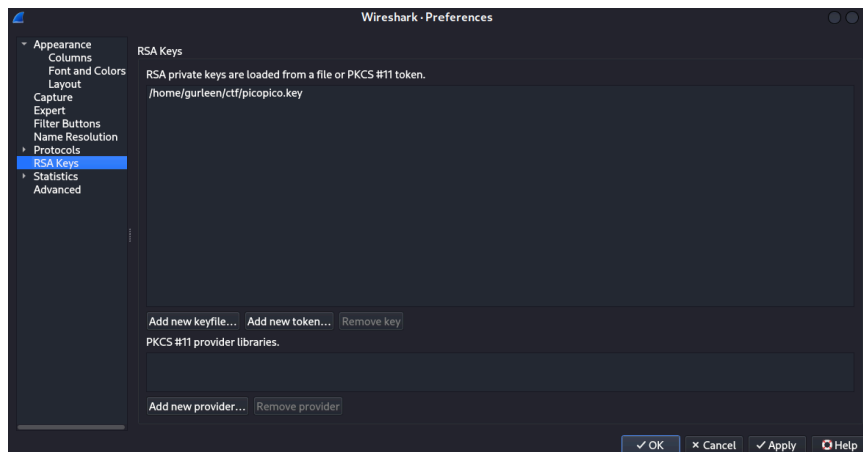
```
(gurleen@kali)-[~/ctf]
$ cat picopico.key
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQCwKlFPNKjseJF5
puCJU5x38XcT1eQge5z0KNahALYudvGV0Es61TnIgvceR4ko8i30Cwak2/atcGk3
oz9jFKep7XFEYNP31IwwD9j/YazlKy4DRLG0b0yIZUU1f2WRA7Uhf0POQXsDT1oU
X32jMKZkQSSDW4MRZd9trJYd02TrcEPMsBiZQLFlvgnNwl3QlawozTHLAJKI36j1
cPwSMMeNca1e0Zi1s7R5IxfhpNX0BF0FmxiWvmeOHbaspyHg8UEmGBrkd4k4wXSK
GQvrc8QjycP4ScEdquxJiYnDT8iEbAq70/7f/5NIN1DE9YoGJqKYjTS9nRPB4Yvj
JN/SJnhvAgMBAAECggEACnd3LrG/TZVH3sR0qvq01CwQPYPfUXdLVyNHab7EWon
pc+XB0HurJENG2CpRYF7h+nQ5ADhfIYSCicBf/jsEB7VueJ20CxEVtHVL3h6R6Bp
oHmle0Em80cofuMpdL/ko+om3T8BkVSzCvCl5NMTUuAF7iRmfX7oDLALwM0IzzQv
2un+2UmT15rgAZfl3IL1PGvJhbbLxfeePE9MBy1SqBjQ9rNFn8sQv959J6BH4b
EpK//ErtNP2yh7oiVBBgKEQ1gEu0jQC/4oxoqCFfZaf9XNRCxB/zY1nUprvJyz09
NMQWNF2EmvMBVGfoTxmuut5N0GbVr2UyHxWMKm2s0QKBgQDpb2+AWgWLGtetuLKJ
fJs8dnd6LhnafbKCOXMOT68qMBRoTpBtVTLRVSNvWcm8m4TTEazX4+ZA+bJFwUFW
aATDmHcr6lMI3tNKrcsnY2F7o5I4z6mwuRuSesZq/ndxZqCzwCu4nKixh3cznp7j
JiElNG0d8Lu5eQgmVAK1AhWfQKBgQDBMa9ga7VJUP4pzcHnWAoi340pfjvQYeGl
IKL3AK040edaHdH9qid41PQHnL703xzN669SkLZ5s0d88A/LFLk4oZNMKdkSTQIQ
+AMBxH01HGFvnCOuPg/FbNp1wS7zJEg5u5HFQWyMPNJLr/hZ6g2Yp+UGpAcGTWm/
RCPVAPhLWwKBgQDAB00aonPavjKGXiHAqBirrgiswA/S5QqrzEaxxys5cUPYaoi0
6BldysPTnJr45JZna2rcTkXjvYTBjTdf3zHMFwGzyBfefC8kh8NPK5nNs8ldorbd
AemEnjBkP+DSELKyK6vLuL0rdtZAQgRCp+MsT+xTb02ArefeX826SXspoQKBgC2v
nDOHBQXje1dTawlUtoFUrgQE8AwLOYEdKKyUoCLOvqEW8D02a0MtyM+MB6tQI7Wm
iH1T73L0LHGLK3bw3aRAwV5/fu/0+jAdFk8AHjPTFE+acu2fi4c6aKb0GjAXYksU
yjiFeK/pKinV4SESMkjpw0WowGiDgtcRPBAA/LaFAoGAfEM1rfM0v3UmB7PS6u0m
P3ckP2CFCdaryXPfC52GBcJ3Q46YpsQvLTVotM+teHvTjNw2jwwZxIL4NenGSEj3
KDhQo0iQC9BrDD+DB4I9+T9nxT3g7R6MrgITghB4We7TVhL/PljnJTyDqpjNA4kY
TveAJPv6Xq1ERT5PutX3BqQ=
-----END PRIVATE KEY-----
```

Step 2: Load the PCAP into Wireshark

Opened `capture.pcap` in Wireshark.

Then configured Wireshark to decrypt the TLS stream using the provided RSA key:

1. Navigated to **Edit → Preferences**
2. Selected **Protocols → TLS**
3. Added the key to the **RSA Keys list** for port 443 traffic



Step 3: Locate the Flag in SSL Stream

With decryption enabled, located a JPEG payload (JFIF marker) in the reassembled SSL packets. In the decoded view, the flag was visible as ASCII inside the JPEG segment.

80 0.611581	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=41627 Win=124416 Len=0 TSval=133588140 TSecr=570160922
90 0.611587	128.237.140.23	172.31.22.220	TCP	66 [TCP Window Update] 57930 → 443 [ACK] Seq=1755 Ack=41627 Win=131072 Len=0 TSval=133588140 TSecr=570160922
91 0.611591	172.31.22.220	128.237.140.23	HTTP	2275 HTTP/1.1 200 OK (JPEG JFIF image)
92 0.639462	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=44375 Win=129664 Len=0 TSval=133588168 TSecr=570160922
93 0.639481	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=45740 Win=131072 Len=0 TSval=133588168 TSecr=570160951
94 0.640260	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=48497 Win=128320 Len=0 TSval=133588168 TSecr=570160951
95 0.640264	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=51245 Win=125568 Len=0 TSval=133588168 TSecr=570160951
96 0.640267	128.237.140.23	172.31.22.220	TCP	66 [TCP Window Update] 57930 → 443 [ACK] Seq=1755 Ack=51245 Win=131072 Len=0 TSval=133588168 TSecr=570160951
97 0.640944	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=53993 Win=128320 Len=0 TSval=133588169 TSecr=570160951
98 0.640948	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=56741 Win=125568 Len=0 TSval=133588169 TSecr=570160952
99 0.640950	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=59489 Win=122616 Len=0 TSval=133588169 TSecr=570160952
100 0.640953	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=62237 Win=120864 Len=0 TSval=133588169 TSecr=570160952
101 0.640955	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=64985 Win=117312 Len=0 TSval=133588169 TSecr=570160952
102 0.641030	128.237.140.23	172.31.22.220	TCP	66 57930 → 443 [ACK] Seq=1755 Ack=67733 Win=128320 Len=0 TSval=133588169 TSecr=570160952


00000100	00 00 00 01 00 00 00 01 00 00 00 01 70 09 63 01pic0
00000108	43 54 46 70 08 0f 0e 05 09 2e 72 0f 03 75 74 08	CTF{h0ne y,roast0
00000109	64 58 70 05 01 06 75 74 13 70 00 0f e2 02 1c	0,000001 01 00 00
000001f0	49 43 43 5f 50 52 4f 40 49 4c 45 00 01 01 00 00	ICC PROF ILE....
00000200	02 0c 0c 03 6d 73 02 10 00 00 6d 0e 74 72 52 47	..lcas.. ..mtrR0
00000210	42 20 58 59 5a 20 07 dc 00 01 00 19 00 03 00 20	B XYZ.....)
00000220	00 39 61 63 73 70 41 50 50 4c 00 00 00 00 00 00	..9acspAP PL.....
00000230	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000240	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000250	6d 73 00 00 00 00 00 00 00 00 00 00 00 00 00 00	ms.....
00000260	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000270	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000280	00 0a 64 65 73 63 00 00 00 fc 00 00 00 5e 63 70	..desc.. ..AcP
00000290	72 74 00 00 01 5c 00 00 00 0b 77 74 70 74 00 00	rt..... .wtpT..
000002a0	01 00 00 00 00 14 62 0b 70 74 00 00 01 7c 00 00	..h... .bk pt..]..
000002b0	00 14 72 58 59 5a 00 00 01 00 00 00 00 14 67 58	..rXYZ.....GX
000002c0	59 5a 00 00 01 a4 00 00 00 14 62 58 59 5a 00 00	YZ..... .bXYZ..
000002d0	01 18 00 00 00 14 72 5a 52 43 00 00 01 cc 00 00rT RC.....
000002e0	00 40 67 54 52 43 00 00 01 cc 00 00 00 40 62 54	..BgTRC..... .bBT
000002f0	52 43 00 00 01 cc 00 00 00 40 64 65 73 63 00 00	RC..... .desc..
00000300	00 00 00 00 00 03 63 32 00 00 00 00 00 00 00 00c2.....
00000310	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000320	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Flag Submitted

```
Wireshark · Value (image-jifl.ifd.value_ascii) · capture.pcap  
picoCTF{honey.roasted.peanuts}
```

The flag was extracted from decrypted SSL content and successfully submitted.

Forensics

 Hard

WebNet1

4,357 solves

94% 