

Zadanie 7

I-UPB: Úvod do počítačovej bezpečnosti

November 29, 2021

1 Úvod

Cieľom zadania je demonštrovať útok na nezabezpečenú aplikáciu s ktorou používateľ komunikuje cez protokol HTTPS a neskôr aplikáciu zabezpečiť pomocou HTTPS protokolu a SSL certifikátu podpísaného známou autoritou.

V tomto prípade bola vytvorená jednoduchá statická webová stránka ktorá bola poskytovaná používateľovi reverznej proxy nginx pre zjednodušenie konfigurácie SSL zabezpečenia.

Registration Form

Name:

Email:

Password:

Figure 1: Webová aplikácia

2 Konfigurácia servera

Spomínaný server pre svoje fungovanie pomocou HTTP nepotrebuje žiadnu špeciálnu konfiguráciu iba skopírovať súbory do adresára ktorý je nadstavený ako default pre statické súbory. Zároveň default otvorený port je 80 ktorý sa štandardne používa pre webové aplikácie. Po nahrati html súboru do priečinka *html* môžeme vidieť našu webovú aplikáciu s formulárom.

2.1 HTTP

Keďže náš server je nakonfigurovaný na HTTP protokol tak nástrojom Wireshark dokážeme odchytať nezašifrovanú komunikáciu.

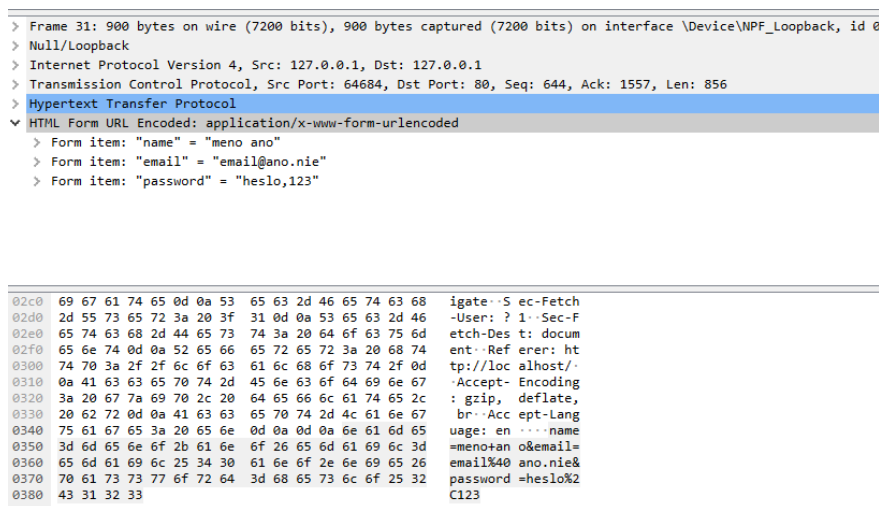


Figure 2: Odchytyvanie nešifrovanej komunikácie

Na obrázku vidíme že dochytené dáta obsahujú nami vyplnený formulára.

2.2 HTTPS

Pokiaľ chceme zabezpečiť komunikáciu aby nemohla byť jednoducho odchytyvaná tak je nutné nginx nadstaviť aby na všetkú prichádzajúcu komunikáciu na porte 80 presmerovával na port 433 a používal HTTPS protokol.

```
server {
    listen      80;
    server_name localhost;
    location / {
        root    html;
        index   index.html index.htm;
    }
    return 301 https://$host$request_uri;
}
```

Uvedená konfigurácia obsahuje defaultné nastavenie adresárov pre statické súbory a hlavne presmerovania každej požiadavky na protokol HTTPS.

Ďalej musíme nakonfigurovať nginx pre HTTPS protokol čo spravíme nasledujúcim konfigom.

```
server {
    listen      443 ssl;
    server_name localhost;
    ssl_certificate      cert.pem;
    ssl_certificate_key  key.pem;
    ssl_session_cache    shared:SSL:1m;
    ssl_session_timeout  5m;
    ssl_ciphers  HIGH:!aNULL:!MD5;
    ssl_prefer_server_ciphers  on;
    location / {
        root    html;
        index   index.html index.htm;
    }
}
```

Konfigurácia obsahuje dôležitú časť a to je SSL certifikát. Použitý SSL certifikát je self-signed čo má za následok že návštevník webovej aplikácie bude najskôr upozornený že certifikát nevydávala žiadna registrovaná certifikačná autorita.

Pre vygenerovanie self-signed SSL certifikátu bol použitý nástroj OpenSSL. Príkaz pre vygenerovanie SSL certifikátu platného 1 rok je:

```
openssl req -x509 -newkey rsa:4096 -nodes -out cert.pem -keyout key.pem -days 365
```

Keď naštívime našu webovú aplikáciu tak sme okmžite presmerovaný na HTTPS protokol ktorý je na porte 433.

Po tejto konfigurácii môžeme odchytať komunikáciu cez HTTPS protokol no neuvidíme žiadne dáta.

18	11.210100	127.0.0.1	127.0.0.1	TCP	44 80 → 64887 [ACK] Seq=2 Ack=2 Win=10233 Len=0
19	11.216325	127.0.0.1	127.0.0.1	TLSv1.2	926 Application Data
20	11.216356	127.0.0.1	127.0.0.1	TCP	44 443 → 64887 [ACK] Seq=2331 Ack=2437 Win=10217 Len=0
21	11.218920	127.0.0.1	127.0.0.1	TLSv1.2	789 Application Data
22	11.218974	127.0.0.1	127.0.0.1	TCP	44 64887 → 443 [ACK] Seq=2437 Ack=3076 Win=10203 Len=0
23	18.294968	127.0.0.1	127.0.0.1	TLSv1.2	75 Encrypted Alert
24	18.294999	127.0.0.1	127.0.0.1	TCP	44 64888 → 443 [ACK] Seq=2 Ack=32 Win=10232 Len=0
25	18.295084	127.0.0.1	127.0.0.1	TCP	44 443 → 64888 [FIN, ACK] Seq=32 Ack=2 Win=10233 Len=0
26	18.295096	127.0.0.1	127.0.0.1	TCP	44 64888 → 443 [ACK] Seq=2 Ack=33 Win=10232 Len=0

Total Length: 922	
Identification: 0xd111 (53521)	
Flags: 0x40, Don't fragment	
...0 0000 0000 0000 = Fragment Offset: 0	
Time to Live: 128	
Protocol: TCP (6)	
Header Checksum: 0x0000 [validation disabled]	
[Header checksum status: Unverified]	
Source Address: 127.0.0.1	
Destination Address: 127.0.0.1	
Transmission Control Protocol, Src Port: 64887, Dst Port: 443, Seq: 1555, Ack: 2331, Len: 882	
Source Port: 64887	
Destination Port: 443	
[Stream index: 0]	

0000	02 00 00 00 45 00 03 9a	d1 11 40 00 80 06 00 00E.....g.....
0010	7f 00 00 01 7f 00 00 01	fd 77 01 bb fb 3b c3 c8w.....
0020	82 d6 77 b7 50 18 27 de	d3 a9 00 00 17 03 03 03	..wP.....
0030	6d 00 00 00 00 00 00 00	05 09 c6 00 fe b3 96 50P
0040	dd f1 24 37 7e 6a 2c c7	4d d8 57 a6 5e f6 08 da	..\$7..M-W-A...
0050	75 eb 90 dc 29 e8 3f c0	50 f8 36 5a 60 c5 d4 5c	u...)? P 6Z...
0060	7c 41 9c 96 52 c2 05 1e	4d b7 8a 6e 80 88 b8 e3	[A-R...R-n...
0070	e7 24 d0 00 71 49 8f 15	4c 14 ae 13 5e 62 f7 40	\$.q...L...b@
0080	ac 0e 8c 20 38 f7 1f 06	20 4e 12 83 ab f2 00 bb	...8...N.....
0090	1f 1c ab 26 25 fa 6c d0	40 77 79 e2 23 6b 9c f3	...&%L @wy #k...
00a0	f9 5b 43 5b d0 7b 63 22	00 7b 7f 76 4e 42 f4 93	[C][c'[-vNB...
00b0	71 c6 9f 09 f7 70 09 47	b1 a5 07 0d e1 f7 b6 86	q...pG.....
00c0	49 f0 20 b7 b3 ec 17 64	41 a5 38 7a b1 fa 1d 51	I-...d A 8z...Q

Total Length (p.len), 2 byte(s)

Figure 3: Odchytávanie šifrovanej komunikácie