

# Úvod do počítačovej bezpečnosti

## Zadanie 7 – TLS (HTTPS) komunikácia s web serverom

Cieľom zadanie bolo vytvoriť bezpečnú komunikáciu medzi užívateľom (browserom) a serverom a otestovať možnosti zneužitia nezabezpečeného, alebo nedostatočne zabezpečeného spojenia.

### A. Server

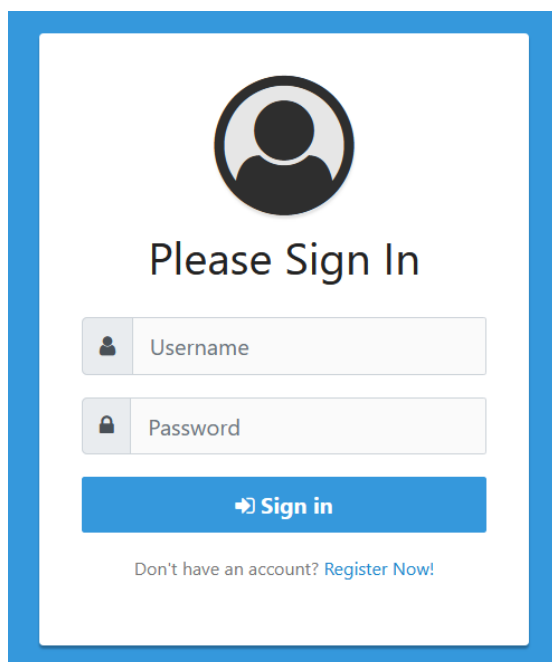
Na konfiguráciu bol použitý takzvaný LAMP Stack

- Operačný systém založený na Linuxe (Ubuntu v mojom prípade)
- Webový server Apache
- Databázový server MySQL
- Programovací jazyk PHP

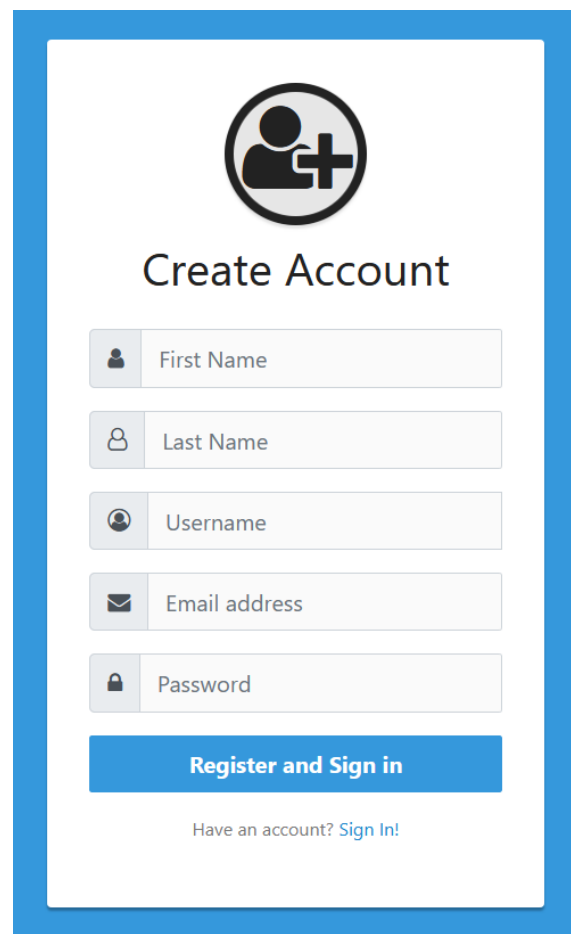
Web server bežal na Linux virtul machine.

### B. Web aplikácia

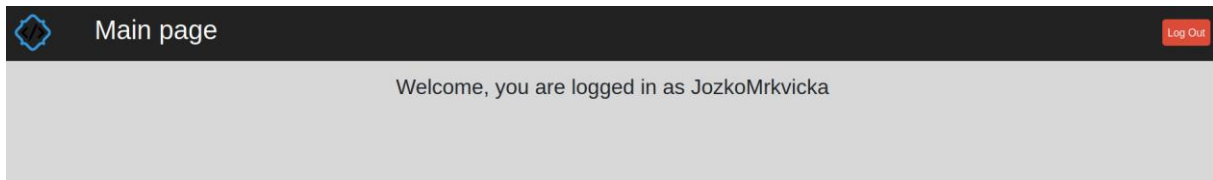
Pre účely tohto zadanie som čiastočne zrecykloval moje zadanie z predmetu webové technológie 2. Aplikácia je tvorená jednoduchým prihlasovacím a registračným formulárom. Po prihlásení je používateľ autentifikovaný a následne presmerovaný na hlavnú stránku ak bola autentifikácia úspešná. Pri registrácii je vytvorený nový užívateľ, ktorý je po úspešnom zaregistrovaní tiež presmerovaný na hlavnú stránku.

The image shows a 'Please Sign In' web form. At the top is a circular icon of a person. Below the icon is the text 'Please Sign In'. There are two input fields: 'Username' with a person icon and 'Password' with a lock icon. Below these is a blue button with a right arrow and the text 'Sign in'. At the bottom, there is a link: 'Don't have an account? Register Now!'.

Obrázok 1: Prihlasovací formulár

The image shows a 'Create Account' web form. At the top is a circular icon of a person with a plus sign. Below the icon is the text 'Create Account'. There are five input fields: 'First Name' with a person icon, 'Last Name' with a person icon, 'Username' with a person icon, 'Email address' with an envelope icon, and 'Password' with a lock icon. Below these is a blue button with the text 'Register and Sign in'. At the bottom, there is a link: 'Have an account? Sign In!'.

Obrázok 2: Registračný formulár



Obrázok 3: Hlavná stránka

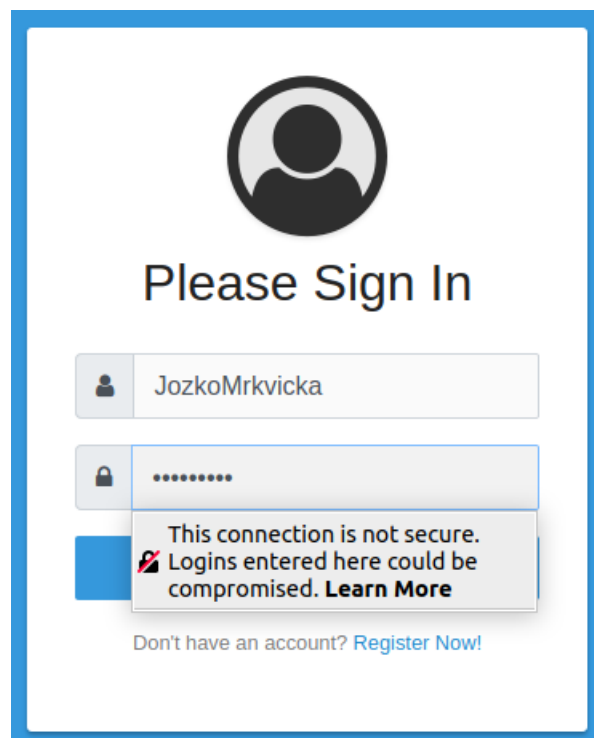
## **C.HTTP konfigurácia**

### **Nastavenie Apache:**

```
<VirtualHost *:80>
    ServerAdmin admin@upb.sk
    ServerName upb.sk
    ServerAlias upb.sk
    DocumentRoot /var/www/upb.sk
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

## **D.Nešifrovaná komunikácia**

Na sledovanie komunikácie bol použitý program Wireshark.



Obrázok 4: Prihlasovací formulár (HTTP)

Pri pokuse o prihlásenie sa na webstránke, ktorá používa len HTTP protokol nás už aj samotný prehliadač varuje pred tým, že naše prihlasovacie údaje môžu byť ohrozené.

Po odoslaní prihlasovacieho formuláru môže aplikácii Wireshark zachytiť POST request, ktorý zodpovedá nášmu pokusu o prihlásenie. Keďže boli naše prihlasovacie údaje odoslané cez POST a nie GET (tak ako by to vždy aj malo byť) musíme ich hľadať pod MIME (Multipurpose Internet Mail Extensions -> prenos textových a netextových príloh).

```

> Frame 120: 1085 bytes on wire (8680 bits), 1085 bytes captured (8680 bits) on interface lo, id 0
> Ethernet II, Src: 00:00:00:00:00:00 (00:00:00:00:00:00), Dst: 00:00:00:00:00:00 (00:00:00:00:00:00)
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
> Transmission Control Protocol, Src Port: 41042, Dst Port: 80, Seq: 1, Ack: 1, Len: 1019
> Hypertext Transfer Protocol
  > POST /php/confirmation.php HTTP/1.1\r\n
    Host: 127.0.0.1.xip.io\r\n
    User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Firefox/83.0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
    Accept-Language: en-US,en;q=0.5\r\n
    Accept-Encoding: gzip, deflate\r\n
    Content-Type: multipart/form-data; boundary=-----56213936917796595242533143317\r\n
  > Content-Length: 426\r\n
    [Content length: 426]
    Origin: http://127.0.0.1.xip.io\r\n
    Connection: keep-alive\r\n
    Referer: http://127.0.0.1.xip.io/\r\n
  > Cookie: PHPSESSID=4drevef6iaqusad7scpel3vct2\r\n
    Upgrade-Insecure-Requests: 1\r\n
    \r\n
    [Full request URI: http://127.0.0.1.xip.io/php/confirmation.php]
    [HTTP request 1/2]
    [Response in frame: 123]
    [Next request in frame: 153]
    File Data: 426 bytes
  MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----56213936917796595242533143317"

```

Obrázok 5: POST

Po rozbalení MIME môžeme medzi dátami jasne vidieť naše prihlasovacie údaje:

**JozkoMrkvicka:papagaj98**

```

> MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----
  [Type: multipart/form-data]
  First boundary: -----56213936917796595242533143317\r\n
  > Encapsulated multipart part:
    Content-Disposition: form-data; name="username"\r\n\r\n
    > Data (13 bytes)
      Data: 4a6f7a6b6f4d726b7669636b61
      [Length: 13]
      Boundary: \r\n-----56213936917796595242533143317\r\n
    > Encapsulated multipart part:
      Content-Disposition: form-data; name="password"\r\n\r\n
      > Data (9 bytes)
        Boundary: \r\n-----56213936917796595242533143317\r\n
    > Encapsulated multipart part:
      Last boundary: \r\n-----56213936917796595242533143317--\r\n

```

0290	0a 0d 0a	2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d	...
02a0	2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d 2d	-----	
02b0	35 36 32 31 33 39 33 36	39 31 37 37 39 36 35 39	56213936 91779659
02c0	35 32 34 32 35 33 33 31	34 33 33 31 37 0d 0a 43	52425331 43317..C
02d0	6f 6e 74 65 6e 74 2d 44	69 73 70 6f 73 69 74 69	ontent-D ispositi
02e0	6f 6e 3a 20 66 6f 72 6d	2d 64 61 74 61 3b 20 6e	on: form -data; n
02f0	61 6d 65 3d 22 75 73 65	72 6e 61 6d 65 22 0d 0a	ame="use rname"...
0300	0d 0a 4a 6f 7a 6b 6f 4d	72 6b 76 69 63 6b 61 0d	..JozkoM rkvetica.
0310	0a 2d 2d 2d 2d 2d 2d 2d	2d 2d 2d 2d 2d 2d 2d 2d	-----
0320	2d 2d 2d 2d 2d 2d 2d 2d	2d 2d 2d 2d 2d 2d 2d 35	-----56
0330	32 31 33 39 33 36 39 31	37 37 39 36 35 39 35 32	21393691 77965952
0340	34 32 35 33 33 31 34 33	33 31 37 0d 0a 43 6f 6e	42533143 317..Con
0350	74 65 6e 74 2d 44 69 73	70 6f 73 69 74 69 6f 6e	tent-Dis position
0360	3a 20 66 6f 72 6d 2d 64	61 74 61 3b 20 6e 61 6d	: form-d ata; nam
0370	65 3d 22 70 61 73 73 77	6f 72 64 22 0d 0a 0d 0a	e="passw ord"....
0380	70 61 70 61 67 61 6a 39	38 0d 0a 2d 2d 2d 2d 2d	papagaj9 8-----
0390	2d 2d 2d 2d 2d 2d 2d 2d	2d 2d 2d 2d 2d 2d 2d 2d	-----
03a0	2d 2d 2d 2d 2d 2d 2d 35	36 32 31 33 39 33 36	----- 56213936
03b0	39 31 37 37 39 36 35 39	35 32 34 32 35 33 33 31	91779659 52425331
03c0	34 33 33 31 37 0d 0a 43	6f 6e 74 65 6e 74 2d 44	43317..C ontent-D
03d0	69 73 70 6f 73 69 74 69	6f 6e 3a 20 66 6f 72 6d	ispositi on: form
03e0	2d 64 61 74 61 3b 20 6e	61 6d 65 3d 22 73 75 62	-data; n ame="sub

Obrázok 6: Nešifrované dáta

## E. HTTPS konfigurácia

Vytvorenie certifikátu cez **Let's Encrypt** ani cez **sslforfree** nebolo v mojom prípade úspešné vzhľadom na to, že ani jeden nevytvorí certifikát na lokálnu IP adresu (samozrejme s použitím xip.io/nip.io) a vzhľadom na to, že server nám poskytnutý nebol a osobne nie som ochotný si kvôli zadaniu nastavovať router tak aby to fungovalo, som sa rozhodol použiť selfsigned certifikát. Výsledné zabezpečenie bude rovnaké len samotný certifikát nebude overený žiadnou dôveryhodnou certifikačnou autoritou takže prehliadač bude ukazovať hlášku, že certifikát nie je overený avšak komunikácia bude normálne zabezpečená a šifrovaná (ak by som si tento certifikát pridal do trust store môjho OS, prehliadač by s nim už nemal žiaden problém).

Viac o problematike vytvárania certifikátov na localhoste a tiež aj prečo je to dokonca nebezpečné robiť cez certifikačné authority je možné si prečítať na tomto linku:

<https://letsencrypt.org/docs/certificates-for-localhost/>

### Vytvorenie TLS/SSL certifikátu (certifikát na použitie s SSL a TLS)

```
$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apacheselfsigned.crt
```

### Nastavenie SSL:

```
SSLCipherSuite ECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH
SSLProtocol All -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
SSLHonorCipherOrder On
Header always set X-Frame-Options DENY
Header always set X-Content-Type-Options nosniff
SSLCompression off
SSLUseStapling on
SSLStaplingCache "shmcb:logs/stapling-cache(150000)"
SSLSessionTickets Off
```

- SSL 2.0, SSL 3.0 a TLS 1.0 sú citlivé na známe útoky a preto sú blokované
- TLS 1.1 je už v dnešnej dobe zastaralý a preto je tiež zakázaný
- SSLCipherSuite – šifrovacie špecifikácie
- SSLHonorCipherOrder – použitie šifrovacích predvolieb servera
- SSLCompression – zakázanie kompresie chráni pred TLS compression oracle útokmi
- SSLUseStapling – umožňuje OCSP stapling (OCSP - Online Certificate Status Protocol - je mechanizmus určujúci, či bol certifikát servera odvolaný alebo nie)
- Zakázanie protokolu SSLSessionTickets zabezpečuje, že funkcia Perfect Forward Secrecy nebude narušená ak sa server nerešartuje pravidelne

### Nastavenie Apache:

```
<VirtualHost *:443>
    ServerAdmin admin@upb.sk
    ServerName upb.sk
    DocumentRoot /var/www/upb.sk
    SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt
    SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
    <FilesMatch "\.(cgi|shtml|phtml|php)$">
        SSLOptions +StdEnvVars
    </FilesMatch>
    <Directory /usr/lib/cgi-bin>
        SSLOptions +StdEnvVars
    </Directory>
</VirtualHost>
```

<b>Subject Name</b>	
Country	SK
State/Province	Bratislava
Locality	Bratislava
Organization	Slovak University of Technology
Organizational Unit	Faculty of Electrical Engineering and Information Technology
Common Name	upb.sk
Email Address	*****@stuba.sk
<b>Validity</b>	
Not Before	11/21/2020, 8:30:55 PM (Central European Standard Time)
Not After	11/21/2021, 8:30:55 PM (Central European Standard Time)
<b>Public Key Info</b>	
Algorithm	RSA
Key Size	2048
Exponent	65537
Modulus	C5:31:B5:0A:9D:F1:76:14:08:61:52:A6:72:15:EE:32:15:BA:3A:93:80:56:E6:02:1F:7D...
<b>Miscellaneous</b>	
Serial Number	06:41:C2:A0:88:D8:82:8B:58:68:46:46:2F:6C:16:1B
Signature Algorithm	SHA-256 with RSA Encryption
Version	3
Download	<a href="#">PEM (cert)</a> <a href="#">PEM (chain)</a>
<b>Fingerprints</b>	
SHA-256	FE:B1:9D:4D:9F:89:79:50:08:9D:72:B5:AD:96:D5:F2:98:B1:31:04:0B:9B:60:C0:78:C...
SHA-1	C6:35:8F:35:22:3B:BA:F2:73:C0:C3:1A:00:A3:09:91:DD:87:C8:36

Obrázok 7: Selfsigned certifikát

## F. Šifrovaná komunikácia

Na sledovanie komunikácie bol použitý program Wireshark.

Tentokrát pri pokuse o prihlásenie sa na webstránke nás už prehliadač nevaruje pred tým, že by naše prihlasovacie údaje mohli byť ohrozené.

Po odoslaní prihlasovacieho formuláru môžeme v aplikácii Wireshark zachytiť **TLS** komunikáciu medzi serverom a klientom (prehliadačom). Klient a server vykonávajú handshake, následne dôjde k výmene šifier. Ak by sme teraz chceli zistiť prihlasovacie údaje už ich uvidíme len v zašifrovanej forme jedine čo môžem aj naďalej vidieť je kto s kým komunikuje.

202 2594.6232834... 127.0.0.1	127.0.0.1	TCP	66 43380 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=915605727...
203 2594.6256998... 127.0.0.1	127.0.0.1	TLSv1.3	583 Client Hello
204 2594.6257305... 127.0.0.1	127.0.0.1	TCP	66 443 → 43380 [ACK] Seq=1 Ack=518 Win=65024 Len=0 TSval=9156057...
205 2594.6390893... 127.0.0.53	127.0.0.1	DNS	87 Standard query response 0x659c AAAA 127.0.0.1.xip.io OPT
206 2594.6545697... 127.0.0.1	127.0.0.1	TLSv1.3	1846 Server Hello, Change Cipher Spec, Application Data, Applicati...
207 2594.6545888... 127.0.0.1	127.0.0.1	TCP	66 43380 → 443 [ACK] Seq=518 Ack=1781 Win=64128 Len=0 TSval=9156...
208 2594.6722652... 127.0.0.1	127.0.0.1	TLSv1.3	146 Change Cipher Spec, Application Data
209 2594.6722797... 127.0.0.1	127.0.0.1	TCP	66 443 → 43380 [ACK] Seq=1781 Ack=598 Win=65536 Len=0 TSval=9156...
210 2594.6724634... 127.0.0.1	127.0.0.1	TLSv1.3	145 Application Data
211 2594.6725052... 127.0.0.1	127.0.0.1	TLSv1.3	145 Application Data

Obrázok 8: TLSv1.3 komunikácia

210 2594.6724634... 127.0.0.1			127.0.0.1	TLSv1.3	145 Application Data
▶ Frame 210: 145 bytes on wire (1160 bits), 145 bytes captured (1160 bits) on interface lo, id 0 ▶ Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00) ▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1 ▶ Transmission Control Protocol, Src Port: 443, Dst Port: 43380, Seq: 1781, Ack: 598, Len: 79 ▶ Transport Layer Security					
▼ TLSv1.3 Record Layer: Application Data Protocol: http-over-tls Opaque Type: Application Data (23) Version: TLS 1.2 (0x0303) Length: 74					
Encrypted Application Data: 97864e9a91ffbc6ff910cc6831431c2a3f0e613d6f63ed22...					
0000	00 00 00 00 00 00 00 00	00 00 00 00 08 00 45 00	.....E.		
0010	00 83 04 f6 40 00 40 06	37 7d 7f 00 00 01 7f 00	...@.@. 7}.....		
0020	00 01 01 bb a9 74 59 69	ff 70 c9 e0 87 72 80 18	...tYi .p...r..		
0030	02 00 fe 77 00 00 01 01	08 0a 36 93 09 10 36 93	...w.... .6...6.		
0040	09 10 17 03 03 00 4a 97	86 4e 9a 91 ff bc 6f f9	.....J. .N....o.		
0050	10 cc 68 31 43 1c 2a 3f	0e 61 3d 6f 63 ed 22 e5	..h1C.*? .a=oc.."		
0060	d8 9f fa b8 ea 39 ee 37	9f 6a 9d 54 70 4d 45 5a	.....9.7 .j.TpMEZ		
0070	34 6f 50 72 c7 e3 66 9e	65 9e e9 9e f6 38 98 8b	4oPr...f. e....8..		
0080	fa d7 8d 80 e5 f9 6e e5	44 39 45 19 85 40 22 fc	.....n. D9E..@".		
0090	a2		.		

Obrázok 9: Zašifrované dáta