



FAKULTA  
INFORMATIKY  
Masarykova univerzita



SITOLA

## Bezpečnost na síti

---

PB156cv

10. května 2019

Michal Zima

# Motivace

*Proč se zajímáme o bezpečnost na síti*

- vaše komunikace na síti je:
  - odposlechnutelná
  - čitelná
  - replikovatelná
  - zneužitelná
  - pozměnitelná
  - podvrhnutelná
  - cenzurovatelná
  - ...

TLS

# TLS

## *Transport Level Security*

- nástupce dnes již prolomeného protokolu SSL (Secure Sockets Layer)
- široce rozšířený univerzální protokol pro zabezpečení síťového provozu
- ve vrstveném modelu se nachází mezi transportní a aplikační vrstvou
- zajišťuje:
  - důvěrnost (šifrování)
  - integritu
  - autentizaci serveru
  - autentizaci klienta

## Struktura TLS rámce

Content type (1 B)	Legacy version (2 B)	Length (2 B)	Protocol message	(MAC (4 B))
--------------------	----------------------	--------------	------------------	-------------

## Stavební kameny

- veřejný klíč serveru podepsaný certifikační autoritou
  - nezbytný pro zahájení komunikace a autentizaci serveru
- certifikát veřejného klíče klienta
- dohodnutý sdílený klíč pro symetrické šifrování dat
- MAC (message authentication code) pro zajištění integrity

# Handshake

- klient požádá server o certifikát
- server certifikát dodá a nabídne sady šifer
- klient si vybere
- ustaví se sdílený klíč

# Handshake

- klient požádá server o certifikát
- server certifikát dodá a nabídne sady šifer
- klient si vybere
- ustaví se sdílený klíč



## s\_client

- nástroj z balíku openssl (volá se `$ openssl s_client`)
- umožňuje testovat konfiguraci TLS na serveru
- je možné ho využít i k líné implementaci podpory TLS v klientovi

## s\_client

### *Hlavní argumenty*

- -CAfile <cert.pem>
- -servername <sni>
- -crlf
- -connect <host>:<port>

# s\_client

## Zadání

1. Stáhněte si <http://147.251.54.177/ca/ca.cert.pem>
2. Pomocí s\_client se připojte na 147.251.54.177:443 se staženým certifikátem CA, s nastaveným SNI na „pb156“, se zapnutými CRLF a wiresharkem odchytněte provoz vyvolaný následujícím požadavkem:

```
GET / HTTP/1.1
```

```
Host: 147.251.54.177
```

3. Zjistěte, jaké typy TLS paketů se v provozu objevují a kde obsahují nešifrovanou informaci o webové stránce, ke které přistupujete.
4. Vyzkoušejte, že je možné se zabezpečeně připojit k SMTP serveru relay.muni.cz:465  
EHL0 pb156cv.fi.muni.cz  
QUIT

## nmap

- nástroj pro skenování sítě/strojů
- kromě standardního skenování zahrnuje i řadu skriptů pro pokročilé skeny, mj. i TLS:

```
$ ls /usr/share/nmap/scripts/ | grep "^ssl\|tls"
```

```
ssl-ccs-injection.nse
```

```
ssl-cert-intaddr.nse
```

```
ssl-cert.nse
```

```
ssl-date.nse
```

```
ssl-dh-params.nse
```

```
ssl-enum-ciphers.nse
```

```
ssl-heartbleed.nse
```

```
ssl-known-key.nse
```

```
ssl-poodle.nse
```

```
sslv2-drown.nse
```

```
sslv2.nse
```

```
tls-alpn.nse
```

```
tls-nextprotoneg.nse
```

```
tls-ticketbleed.nse
```

# nmap

## Zadání

1. Zjistěte, jaké sady šifer nabízí server 147.251.54.177:443:  
`nmap --script ssl-enum-ciphers -p 443 147.251.54.177`
2. Provéřte také server `idp2.ics.muni.cz` a popište, jaké problémy má jeho konfigurace.

# Tunelování

# stunnel

- nástroj pro vytvoření TLS proxy
- přidává podporu pro TLS klientům i serverům
- nevyžaduje žádné zásahy do původních programů

# stunnel

## Zadání

1. Zkonfigurujte stunnel (/etc/stunnel/stunnel.conf) následujícím způsobem:

- přepněte stunnel do režimu klienta
- vytvořte sekci fimuni
- nechte stunnel naslouchat na localhost:8080 (nebo 127.0.0.1:8080)
- nechte stunnel se připojovat k www.fi.muni.cz:443
- Tip: vzorový konfigurační soubor je v /usr/share/doc/stunnel4/examples/

2. Připojte se ke svému tunelu pomocí nástroje netcat:

```
nc -C localhost 8080
```

```
GET / HTTP/1.1
```

```
Host: www.fi.muni.cz
```

3. Pozorujte wiresharkem, k jakému problému dojde, pokud se pokusíte k tunelovanému serveru přistoupit pomocí nástroje wget:

```
wget http://localhost:8080/
```



## SSH tunelování

- kromě zabezpečeného shellu, zprostředkování X11 nebo přenosu souborů umožňuje i přeposílání libovolného TCP provozu – tzv. SSH tunelování
- využitelné zejména pro zabezpečený přenos tam, kde služba zabezpečení nepodporuje
- lze využít i pro dočasné zpřístupnění služeb dostupných pouze z vnitřní sítě
- tunel může být vytvořen lokálně nebo vzdáleně

# SSH tunelování

## *Klíčové argumenty*

-L [bind\_address:]port:host:hostport

- vytvoří naslouchací konec tunelu na lokální adrese `bind_address` a TCP portu `port`
- kdykoli přijde nové spojení na lokální konec, přeneseme se na vzdálený server, odkud se otevře spojení na server `host` a port `hostport`
- IPv6 adresy musí být v hranatých závorkách

-N

- nespouští se na vzdálené straně žádný příkaz, neotevírá se shell

-f

- pošle ssh proces na pozadí

# SSH tunelování

## Zadání

1. Vytvořte tunel z lokálního stroje (port 2525) na relay.muni.cz (port 25) přes server aisa
2. Ověřte nástrojem netcat, že je tunel funkční

# SSH tunelování

## *Klíčové argumenty*

-R [bind\_address:]port:host:hostport

- vytvoří naslouchací konec tunelu na vzdálené adrese bind\_address a TCP portu port
- kdykoli přijde nové spojení na vzdálený konec, přeneseme se na lokální stranu, odkud se otevře spojení na server host a port hostport
- IPv6 adresy musí být v hranatých závorkách

# SSH tunelování

## Zadání

1. Vytvořte tunel ze serveru aisa (port 2525) na relay.muni.cz (port 25) přes lokální stroj
2. Ověřte nástrojem netcat, že je tunel funkční