

Petr Krčmář



# *HTTPS na virtuálních web serverech*

*5. listopadu 2011  
LinuxAlt*

IP adres je málo

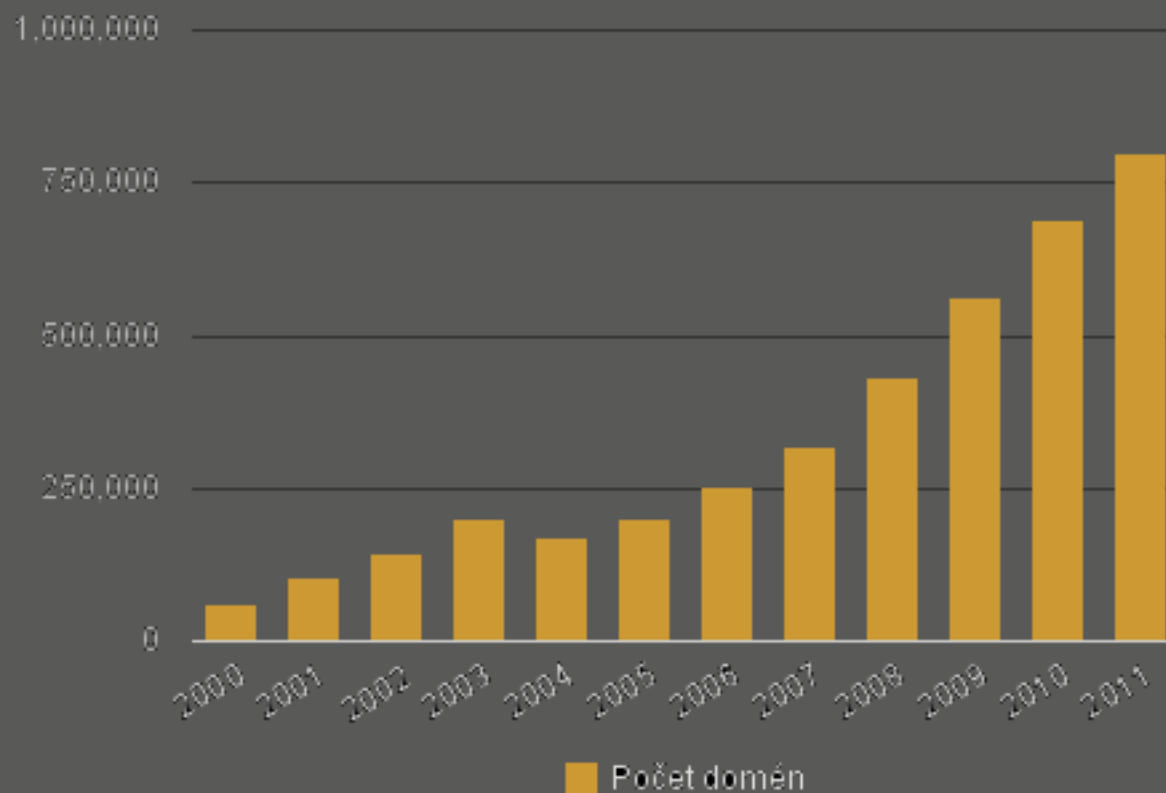


**IPv4**



# Domén naopak přibývá

- Přes 860 000 jen v .CZ



# Výsledek: virtuály + VPS s jednou IP

- Webhosting = stovky domén na jedné IP
- VPS = jedna IP adresa na virtuál
- IPv6 je stále nerozšířené



# Potřeba SSL roste

- Zvyšuje se počet útoků
- Nezabezpečená spojení (free Wi-Fi)
- Je třeba nasadit SSL, ale...



# Na virtuálu nelze mít SSL



## Dotaz: Virtuální SSL server v apache

3.1.2006 19:52 [vasek125](#) | skóre: 30 | Mladá Boleslav

Virtuální SSL server v apache

Přečteno: 211x

[Odpovědět](#) | [Admin](#)



Jak udělám několik z **virtuálních serverů SSL**? Když zapnu SSLengine u prvního, aplikuje se na všechny, když ne, neaplikuje se vůbec, potřebuju prostě několik virtuálních SSL serverů s vlastními certifikáty.

můj prozatimní konfigurák bez ssl konfigurák:  
co mám doplnit?

3.1.2006 19:56 [Michal Kubeček](#) | skóre: 67 | Luštěnice

Re: Virtuální SSL server v apache

[Odpovědět](#) | [Sbalit](#) | [Link](#) | [Blokovat](#) | [Admin](#)

Name based virtuální servery v případě SSL jsou de facto **nerealizovatelné**, protože informace podstatná pro rozlišení virtuálu (**položka Host** v hlavičce dotazu) by byla už v **zašifrované části komunikace**. Takže by byla potřeba vybrat klíč pro komunikaci dřív, než by se server dozvěděl, který má vlastně použít. Někdy se to obchází společným certifikátem (a klíčem) pro všechna jména, ale moc vhodné řešení to není. Resumé: posadte servery na **různé IP adresy** (nebo porty) a použijte IP-based virtuální servery.

Problém?



# Proč to tak je?

- Webservice potřebuje znát jméno webu
- To je předáváno s požadavkem

příklad:

```
GET / HTTP/1.1
```

```
Host: www.root.cz
```

# Příklad z Wiresharku



```
▼ Hypertext Transfer Protocol
  ▸ GET / HTTP/1.1\r\n
    Host: www.root.cz\r\n
    Connection: keep-alive\r\n
    User-Agent: Mozilla/5.0 (X11; Linux i686) AppleWebKit/535.2 (KHTML, like Gecko)
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
    Accept-Encoding: gzip,deflate,sdch\r\n
    Accept-Language: cs-CZ,cs;q=0.8\r\n
    Accept-Charset: UTF-8,*;q=0.5\r\n
```

# Při SSL je ale vše zašifrováno

- Včetně hlavičky Host
- Ta jde až v šifrovaném proudu – to je pozdě
- Server neví, který certifikát chceme
- Není možné si vybírat
- **Je možné použít jen jeden certifikát na IP**
- (Pozn.: problém jen u webu, jinde je STARTTLS – třeba IMAP, POP, SMTP...)

# Příklad spojení na HTTPS



Protocol	Info
TCP	55350 > https [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_
TCP	55351 > https [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_
TCP	https > 55350 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=
TCP	55350 > https [ACK] Seq=1 Ack=1 Win=14608 Len=0 TSval=16
TCP	https > 55351 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=
TCP	55351 > https [ACK] Seq=1 Ack=1 Win=14608 Len=0 TSval=16
TLSv1	Client Hello
TLSv1	Client Hello
TCP	https > 55350 [ACK] Seq=1 Ack=394 Win=6912 Len=0 TSval=4
TLSv1	Server Hello
TCP	55350 > https [ACK] Seq=394 Ack=1449 Win=17504 Len=0 TSv
TCP	https > 55351 [ACK] Seq=1 Ack=394 Win=6912 Len=0 TSval=4
TCP	[TCP segment of a reassembled PDU]
TCP	55350 > https [ACK] Seq=394 Ack=2897 Win=20400 Len=0 TSv

# Řešení?

- Certifikáty to neřeší – různí vlastníci domén
- Řešením by bylo odeslat Host před SSL
- Pak by bylo možné vybrat certifikát
- Nic takového ale ve standardu není
- Je třeba novou vlastnost doplnit
- Řešení existuje a jmenuje se SNI
- (Server Name Indication – oznámení jména serveru)



# Jak to funguje?

- Do Client Hello se přidá server\_name
- Pokud server ví, může zareagovat

TLSv1	Client Hello
TLSv1	Client Hello
TCP	https > 55350 [A
TLSv1	Server Hello
TCP	55350 > https [A

Compression Methods Length: 2
▶ Compression Methods (2 methods)
Extensions Length: 238
▼ Extension: server_name
Type: server_name (0x0000)
Length: 18
Data (18 bytes)
▶ Extension: renegotiation info

12 00 10 00 00 0d 6d 61 69 6c 2e 69 69 6e 66 6f	. . . . .ma il.iinfo
2e 63 7a ff 01 00 01 00 00 0a 00 08 00 06 00 17	.cz. . . . .
00 18 00 19 00 0b 00 02 01 00 00 23 00 b0 52 1b	. . . . . #.R.
76 a2 b3 a9 27 c4 23 52 47 92 b5 89 60 a5 49 08	v...'.#R G...`.I.

# Podpora v serverech

- Apache od verze 2.2.12
- Lighttpd od verze 1.4.24
- Nginx
- Apache Tomcat na Java 7
- Microsoft IIS 8
- LiteSpeed od verze 4.1
- a některé další



# Podpora v prohlížečích

- Firefox od 2.0
- Opera od 8.0
- Chrome
- Safari od 2.1
- Konqueror od 4.7
- Internet Explorer ve Vista a 7
  - v XP není v knihovnách podpora



čili pohoda

# Problém tedy jen...

- Problém tedy jen na IE v XP



Otázka za mikinu:

Kolika procent uživatelů se to tedy dotkne?

Správná odpověď zní

17 %

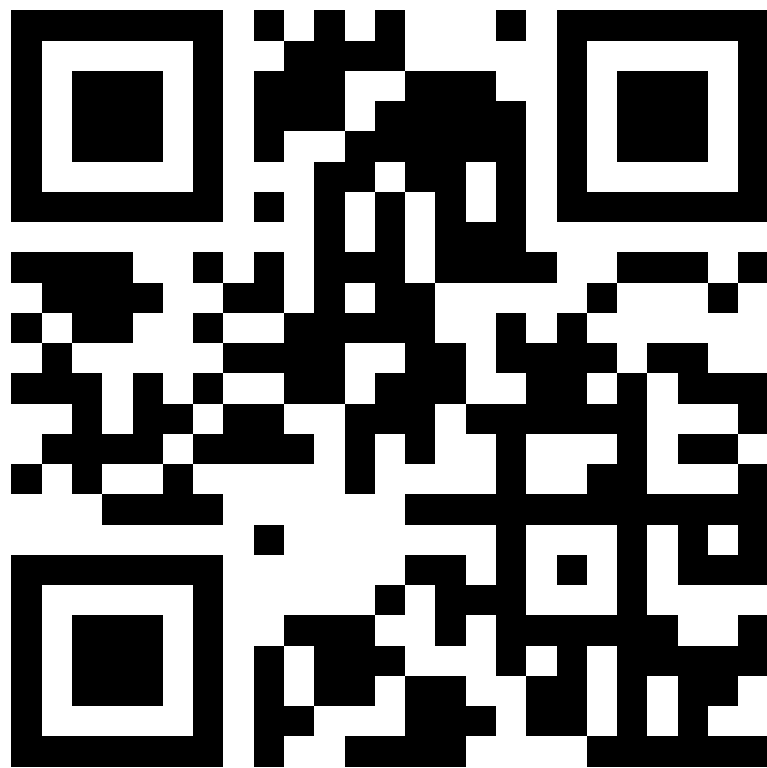
(zdroj: Navrcholu.cz)

# Podpora v mobilních prohlížečích

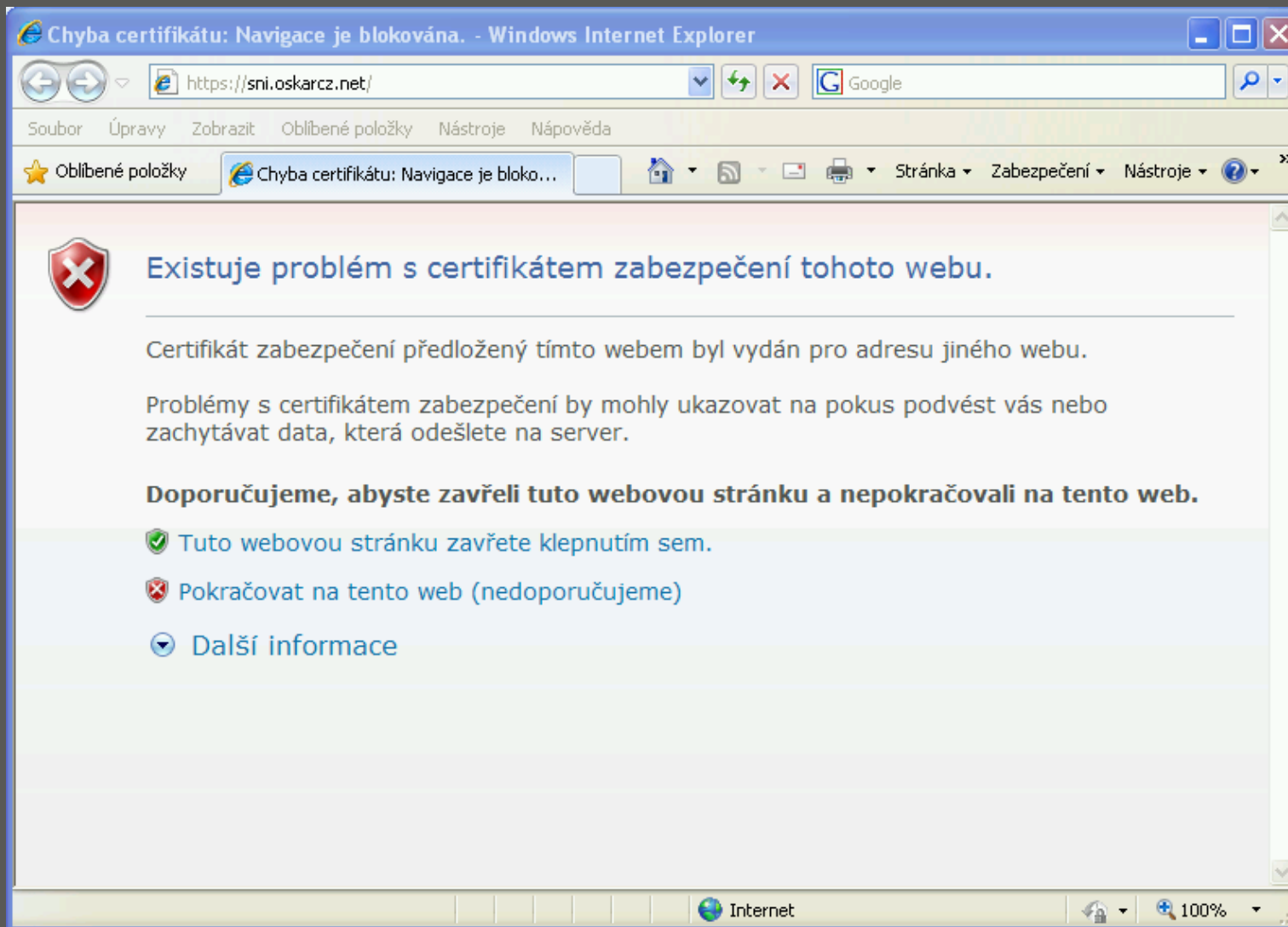
- iOS od verze 4.0 (asi)
- Symbian SNI vůbec neumí
- Android
  - plná podpora od verze 3.0 Honeycomb
  - nižší systémy umí, záleží na prohlížeči
    - Integrovaný ne
    - Opera Mobile ne, Mini neřeší SSL chyby
    - Firefox ano
    - Dolphin ne



# Vyzkoušejte svůj mobil



# Co když to prohlížeč neumí?



# Nasazení Lighttpd

```
$SERVER["socket"] == ":443" {  
    ssl.engine = "enable"  
    ssl.pemfile =  
"/etc/lighttpd/vychozicertifikat.pem"  
    ssl.ca-file = "/etc/lighttpd/ca-  
chain.pem"  
}
```

```
$HTTP["host"] == "www.domena1.cz" {  
    server.document-root =  
"/var/www/domena1.cz"  
    ssl.pemfile =  
"/etc/lighttpd/certifikat1.pem"
```

```
}  
$HTTP["host"] == "www.domena2.cz" {  
    server.document-root =
```



**ROOT.CZ**

# Nasazení Apache

```
# Poslouchej na správném portu a na všech IP
Listen 443
NameVirtualHost *:443
```

```
# Přijímej i uživatele bez SNI
SSLStrictSNIVHostCheck off
```

```
<VirtualHost *:443>
    DocumentRoot /var/www/domena1.cz
    ServerName www.domena1.cz
    GnuTLSCertificateFile /etc/apache2/certifikat1.pem
    GnuTLSKeyFile /etc/apache2/klic1.key
</VirtualHost>
```

```
<VirtualHost *:443>
    DocumentRoot /var/www/domena2.cz
    ServerName www.domena2.cz
    GnuTLSCertificateFile /etc/apache2/certifikat2.pem
    GnuTLSKeyFile /etc/apache2/klic2.key
</VirtualHost>
```

Dotazy?



Děkuji za pozornost



*Petr Krčmář*

*www.root.cz, www.debian-linux.cz*

*petr.krcmar@iinfo.cz*

*GPG: 9FBEA4F5*

Petr Krčmář



*HTTPS na virtuálních  
web serverech*

*5. listopadu 2011  
LinuxAlt*



IP adres je málo



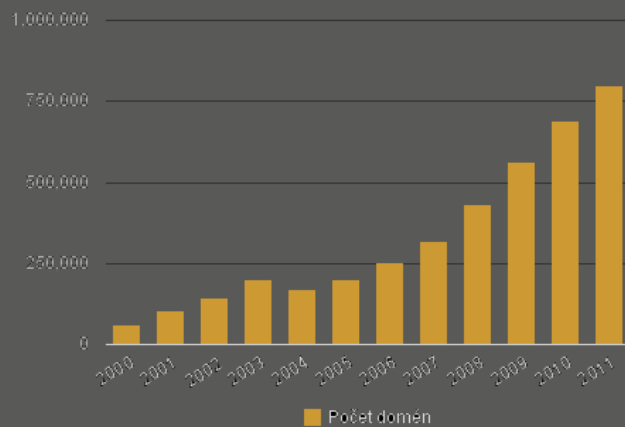
# IPv4



 **ROOT.CZ**

# Domén naopak přibývá

- Přes 860 000 jen v .CZ



 **ROOT.CZ**

## Výsledek: virtuály + VPS s jednou IP

- Webhosting = stovky domén na jedné IP
- VPS = jedna IP adresa na virtuál
- IPv6 je stále nerozšířené



## Potřeba SSL roste

- Zvyšuje se počet útoků
- Nezabezpečená spojení (free Wi-Fi)
- Je třeba nasadit SSL, ale...



# Na virtuálu nelze mít SSL



## Dotaz: Virtuální SSL server v apache

3.1.2006 19:52 [vasek125](#) | skóre: 30 | Mladá Boleslav

Virtuální SSL server v apache

Přečteno: 211x

[Odpovědět](#) | [Admin](#)



Jak udělám několik z **virtuálních serverů SSL?** Když zapnu SSLengine u prvního, aplikuje se na všechny, když ne, neaplikuje se vůbec, potřebuju prostě několik virtuálních SSL serverů s vlastními certifikáty.

můj prozatímní konfigurák bez ssl konfigurák:  
co mám doplnit?

3.1.2006 19:56 [Michal Kubeček](#) | skóre: 67 | Luštěnice

Re: Virtuální SSL server v apache

[Odpovědět](#) | [Sbalit](#) | [Link](#) | [Blokovat](#) | [Admin](#)

Name based virtuální servery v případě SSL jsou de facto **nerealizovatelné**, protože informace podstatná pro rozlišení virtuálu (**položka Host** v hlavičce dotazu) by byla už v **zašifrované části komunikace**. Takže by bylo potřeba vybrat klíč pro komunikaci dřív, než by se server dozvěděl, který má vlastně použít. Někdy se to obchází společným certifikátem (a klíčem) pro všechna jména, ale moc vhodné řešení to není. Resumé: posadte servery na **různé IP adresy** (nebo porty) a použijte IP-based virtuální servery.



Problém?



(říct vtíp o čekárně)

## Proč to tak je?

- Webserver potřebuje znát jméno webu
- To je předáváno s požadavkem  
příklad:

```
GET / HTTP/1.1
```

```
Host: www.root.cz
```



# Příklad z Wiresharku



```
▼ Hypertext Transfer Protocol
  ▸ GET / HTTP/1.1\r\n
    Host: www.root.cz\r\n
    Connection: keep-alive\r\n
    User-Agent: Mozilla/5.0 (X11; Linux i686) AppleWebKit/535.2 (KHTML, like Gecko)
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
    Accept-Encoding: gzip,deflate,sdch\r\n
    Accept-Language: cs-CZ,cs;q=0.8\r\n
    Accept-Charset: UTF-8,*;q=0.5\r\n
```



## Při SSL je ale vše zašifrováno



- Včetně hlavičky Host
- Ta jde až v šifrovaném proudu – to je pozdě
- Server neví, který certifikát chceme
- Není možné si vybírat
- **Je možné použít jen jeden certifikát na IP**
- (Pozn.: problém jen u webu, jinde je STARTTLS – třeba IMAP, POP, SMTP...)



# Příklad spojení na HTTPS



Protocol	Info
TCP	55350 > https [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK
TCP	55351 > https [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK
TCP	https > 55350 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=
TCP	55350 > https [ACK] Seq=1 Ack=1 Win=14608 Len=0 TSval=16
TCP	https > 55351 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=
TCP	55351 > https [ACK] Seq=1 Ack=1 Win=14608 Len=0 TSval=16
TLSv1	Client Hello
TLSv1	Client Hello
TCP	https > 55350 [ACK] Seq=1 Ack=394 Win=6912 Len=0 TSval=4
TLSv1	Server Hello
TCP	55350 > https [ACK] Seq=394 Ack=1449 Win=17504 Len=0 TSv
TCP	https > 55351 [ACK] Seq=1 Ack=394 Win=6912 Len=0 TSval=4
TCP	[TCP segment of a reassembled PDU]
TCP	55350 > https [ACK] Seq=394 Ack=2897 Win=20400 Len=0 TSv

# Řešení?



- Certifikáty to neřeší – různí vlastníci domén
- Řešením by bylo odeslat Host před SSL
- Pak by bylo možné vybrat certifikát
- Nic takového ale ve standardu není
- Je třeba novou vlastnost doplnit
- Řešení existuje a jmenuje se SNI
- (Server Name Indication – oznámení jména serveru)



# Jak to funguje?

- Do Client Hello se přidá server\_name
- Pokud server ví, může zareagovat

```
TLStv1 Client Hello
TLStv1 Client Hello
TCP https > 55350 [A
TLStv1 Server Hello
TCP 55350 > https [A
```

```
Compression Methods Length: 2
  Compression Methods (2 methods)
Extensions Length: 238
  Extension: server_name
    Type: server_name (0x0000)
    Length: 18
    Data (18 bytes)
  Extension: renegotiation info
```

```
12 00 10 00 00 0d 6d 61 69 6c 2e 69 69 6e 66 6f .....ma il.iinfo
2e 63 7a ff 01 00 01 00 00 0a 00 08 00 06 00 17 .CZ.....
00 18 00 19 00 0b 00 02 01 00 00 23 00 b0 52 1b .....#...R.
76 a2 b3 a9 27 c4 23 52 47 92 b5 89 60 a5 49 08 v...'#R G...I.
```

## Podpora v serverech

- Apache od verze 2.2.12
- Lighttpd od verze 1.4.24
- Nginx
- Apache Tomcat na Java 7
- Microsoft IIS 8
- LiteSpeed od verze 4.1
- a některé další



## Podpora v prohlížečích

- Firefox od 2.0
- Opera od 8.0
- Chrome
- Safari od 2.1
- Konqueror od 4.7
- Internet Explorer ve Vista a 7
  - v XP není v knihovných podpora



čili pohoda

## Problém tedy jen...

- Problém tedy jen na IE v XP



Otázka za mikinu:  
Kolika procent uživatelů se to tedy dotkne?

 **ROOT.CZ**

Správná odpověď zní



17 %

(zdroj: Navrcholu.cz)



# Podpora v mobilních prohlížečích

- iOS od verze 4.0 (asi)
- Symbian SNI vůbec neumí
- Android
  - plná podpora od verze 3.0 Honeycomb
  - nižší systémy umí, záleží na prohlížeči
    - Integrovaný ne
    - Opera Mobile ne, Mini neřeší SSL chyby
    - Firefox ano
    - Dolphin ne



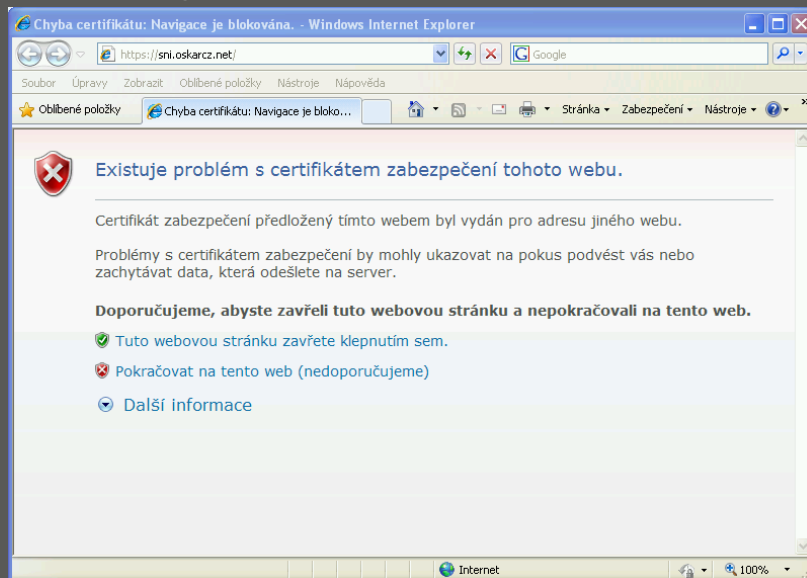
Vyzkoušejte svůj mobil



 **ROOT.CZ**

<https://bob.sni.velox.ch/>

# Co když to prohlížeč neumí?




# Nasazení Lighthttpd



```
$SERVER["socket"] == ":443" {  
    ssl.engine = "enable"  
    ssl.pemfile =  
"/etc/lighttpd/vychozicertifikat.pem"  
    ssl.ca-file = "/etc/lighttpd/ca-  
chain.pem"  
}
```

```
$HTTP["host"] == "www.domena1.cz" {  
    server.document-root =  
"/var/www/domena1.cz"  
    ssl.pemfile =  
"/etc/lighttpd/certifikat1.pem"
```

```
}  
 $HTTP["host"] == "www.domena2.cz" {  
    server.document-root =
```

# Nasazení Apache

```
# Poslouchej na správném portu a na všech IP
Listen 443
NameVirtualHost *:443

# Přijímej i uživatele bez SNI
SSLStrictSNIVHostCheck off

<VirtualHost *:443>
  DocumentRoot /var/www/domena1.cz
  ServerName www.domena1.cz
  GnuTLSCertificateFile /etc/apache2/certifikat1.pem
  GnuTLSKeyFile /etc/apache2/klic1.key
</VirtualHost>

<VirtualHost *:443>
  DocumentRoot /var/www/domena2.cz
  ServerName www.domena2.cz
  GnuTLSCertificateFile /etc/apache2/certifikat2.pem
  GnuTLSKeyFile /etc/apache2/klic2.key
</VirtualHost>
```



Dotazy?



 ROOT.CZ

Děkuji za pozornost



*Petr Krčmář*

*www.root.cz, www.debian-linux.cz*

*petr.krcmar@iinfo.cz*

*GPG: 9FBEA4F5*

