TLS certifikát pod mikroskopem

Petr Krčmář

3. května 2016

Prezentace už teď na webu

https://www.petrkrcmar.cz

Proč nasazovat HTTPS?

- HTTPS není jen pro banky a mail
- autenticita přenášených dat
- odposlech je považován za útok (RFC 7258)
- pozměňování přenosů, supercookies, malware
- ochrana osobních údajů
- zabránění únosu session cookie
- možnost nasazení HTTP/2 – výkon

Proč nasazovat HTTPS?

- HTTPS není jen pro banky a mail
- autenticita přenášených dat
- odposlech je považován za útok (RFC 7258)
- pozměňování přenosů, supercookies, malware
- ochrana osobních údajů
- zabránění únosu session cookie
- možnost nasazení HTTP/2 – výkon
- SSH je normální, proč používat telnet?

Problém: důvěryhodné předání klíče

- šifrovat asymetrickou šifrou umíme
- autentizace stejně důležitá jako silná šifra
- jinak hrozí útok man-in-the-middle
- problém důvěryhodného předání veřejného klíče
- protistrana je pro nás neznámá
- nastupují autority: důvěryhodní prostředníci
- ověří žadatele, vystaví certifikát
- certifikát je **veřejný dokument**

Řetězec důvěry

- software zná kořenové certifikáty
- od serveru dostane řetězec
- certifikáty se musí vzájemně potvrzovat
- konečný certifikát potvrzuje identitu
- zároveň obsahuje veřejný klíč
- identita je potvrzena, klíč předán
- komunikace může začít
- existuje asi 1000 důvěryhodných CA
- většina delegovaných – reálně desítky firem

Co je certifikát

- strojově čitelný dokument ve formátu ASN.1
- datová struktura klíč – hodnota
- každá položka má číselný identifikátor
- a příznak o kritičnosti – nekritičnosti
- kritická nerozpoznaná položka – **musí** být zamítnuta
- nekritická nerozpoznána – **může** být ignorována
- každá rozpoznaná položka **musí** být validována
- certifikát kódován pomocí DER nebo base64 (PEM)

Získání certifikátu

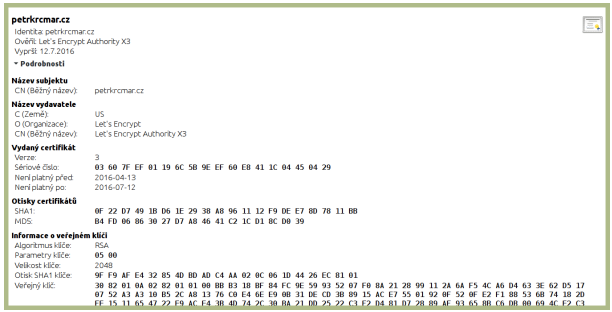
```
$ openssl s_client -showcerts -connect www.petrkrcmar.cz:443 \\
-servername petrkrcmar.cz < /dev/null | openssl x509 -outform PEM

-----BEGIN CERTIFICATE-----
MIIFIjCCBAqgAwIBAgISA2B/7wEZbFue72DoQRwERQQpMA0GCSqGSIb3DQEBCwUA
MEoxCzAJBgNVBAYTAlVTMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MSMwIQYDVQQD
ExpMZXQncyBFbmNyeXB0IEF1dGhvcml0eSBYMzAeFw0xNjA0MTMxMTA3MDBaFw0x
NjA3MTIxMTA3MDBaMBgxFjAUBgNVBAMTDXBldHJrcmNtYXIuY3owggEiMA0GCSqG
SIb3DQEBAQUAA4IBDwAwggEKAoIBAQC7sxi/hPyeWZNSB/CKISiZESpq9Uym1GM+
YtUXB1KjoxC1LKgTdsDkbukLMd7NO4kVrOdVAZIPUg/i8YhTa3QYLf4VEWVHIums
5DtNdCwwuiHdJSLD8tSB1yiJr5Nli8bbAGlM4sM9QrL3BGP1N5SD3h3el2J0Q0r2
920xoT3vDCUWnKvViuggLLSs2sogKrlpKBwCUmvsg2mmgMwbBEIjapg0lUAs7bNa
dn61aOOeN3Y5L2Z5PTusROe8Ib7VISibG4Awo8hjaIpPViwQyFXT/32AZbRxhnFK
arLzk/eFpt5XgfkMeCiGJ/UZ5TWfuxBh9+O8QN4+vdvnh14V2KD1AgMBAAGjggIy
…

WIxNxiCXe/EBWMB04Aek19I+ovnCFnkfsDO5vxDBZYZfumg2YiTwGY2sG/EJNfUx
NsjkxbD7VZuZhJUY40IRm6Iko2QPmQ==
-----END CERTIFICATE-----
```

Prohlížení certifikátu

- ve webovém prohlížeči
- přímo pomocí OpenSSL

```
$ openssl x509 -in certificate.crt -text -noout
```

- gcr-viewer v balíčku gnome-keyring



Co certifikát obsahuje

- Číslo verze
- Sériové číslo
- ID algoritmu podpisu
- Jméno vystavovatele
- Platnost: od do
- Jméno subjektu
- Klíč subjektu
  - algoritmus veřejného klíče
  - veřejný klíč samotný
- Rozšíření a volitelné položky
- Podpis certifikátu
  - algoritmus podpisu
  - podpis samotný

Číslo verze

- verze standardu X.509
- verze formátu certifikátu
- první v RFC 1422
- verze 2 přidala možnost ID autority
- pro případ recyklace jména – nedoporučuje se
- verze 3 – RFC 2459
- přidala volitelná rozšíření
- dnes nejpoužívanější je verze 3

Sériové číslo

- unikátní číslo certifikátu
- v rámci autority se nesmí opakovat
- jméno autority a sériové číslo jednoznačně identifikuje certifikát
- používá se pro identifikaci při revokaci
- jedná se o celé číslo
- není definováno, jak je číslo tvořeno
- klienti musí být schopni zpracovat alespoň 128 bitů
- třeba 03:60:7f:ef:01:19:6c:5b:9e:ef

Algoritmus podpisu

- algoritmus, kterým je podepsán certifikát
- uložen jako identifikátor (OID) dané kombinace
- vždy dvojice kryptografické a hashovací funkce
- historicky RSA a DSA s MD5 a SHA-1
- dnes nepodporované kombinace – nebezpečné
- dnes pouze SHA – až do 512
- podporu ECDSA zavádí RFC 5758
- třeba sha256WithRSAEncryption

Jméno vystavovatele

- neprázdné popisné jméno autority
- obvykle v UTF8
- může obsahovat další položky
  - stát, organizace a další (viz další slide)
  - totéž pak u jména subjektu
- musí odpovídat důvěryhodné autoritě
- přes jména vede řetězec důvěry (!)
- třeba C=US, O=Let’s Encrypt, CN=Let’s Encrypt Authority X3

… formát jména (RFC 5280)

- stát (countryName, C)
- organizace (organizationName, O)
- organizační jednotka (organizationalUnitName, OU)
- stát či oblast (stateOrProvinceName, ST)
- běžné jméno (commonName, CN)

Platnost

- datum a čas – začátek a konec platnosti
- časová zóna Greenwich Mean Time (Zulu)
- včetně sekund, i kdyby byly rovny nula
- do roku 2050 se používá formát UTCTime
  - YYMMDDHHMMSSZ
  - znak Z označuje formát zápisu
  - YY < 50 = 19xx; YY > 50 = 20xx
- od roku 2050 povinně GeneralizedTime
  - YYYYMMDDHHMMSSZ

Jméno subjektu

- subjekt, kterému patří veřejný klíč
- může být tady nebo v rozšíření subjectAltName
- pokud je jen v rozšíření, hlavní jméno je prázdné
- rozšíření pak musí být kritické
- pokud je subjekt CA, pod tímto jménem vydává certifikáty
- třeba CN=petrkrcmar.cz
- u mailu CN=petr.krcmar@iinfo.cz/emailAddress=petr.krcmar@iinfo.cz

Klíč subjektu

- samotný „náklad“ v certifikátu
- nejprve algoritmus klíče (už jsme viděli)
- třeba rsaEncryption (2048 bit)
- poté klíč samotný

```
00:e6:6f:18:0b:2a:c8:c1:e0:cb:9c:dc:67:5c:be:
37:1c:42:96:ab:0b:85:9a:cf:45:5e:0e:f1:39:f5:
a4:c2:92:b4:39:05:c1:25:07:64:a8:77:6d:01:ae:
e0:01:03:76:f3:d4:75:30:f1:b8:8f:e7:7d:5f:9d:
cb:25:df:8f:44:c1:7d:6c:c4:b8:b3:1b:23:75:70:
1f:d9:ef:62:3f:2f:f7:91:dc:e3:07:9f:67:37:f7:
0c:51:85:f7:3e:e3:17:04:8a:31:21:11:c5:64:c8:
36:15:5a:a1:57:41:c0:4f:26:f1:71:16:5d:39:25:
4a:a1:85:fc:48:45:0c:83:c8:ae:b1:aa:67:3f:81:
f6:e6:ff:98:2b:34:73:ae:e9:07:01:4b:ac:18:e0:
…
```

Rozšíření dle RFC 5280

- součástí certifikátu jsou volitelná rozšíření
- vždy mají označení kritičnosti (výchozí je FALSE)
- umožňují doplnit informace o užití certifikátu
- každé rozšíření se může objevit jen jednou
- CA musí některá rozšíření povinně uvést
  - key identifiers
  - basic constraints
  - key usage
  - certificate policies

Důležitá rozšíření

- Basic Constraints – patří certifikát CA?
- Subject Alternative Name – další identity subjektu
- CRL Distribution Points – odkazy pro CRL (viz revokace)
- Authority Information Access – odkaz pro OCSP (viz revokace)
- Authority Key Identifier – identifikace klíče pro podpis revokací
- Subject Key Identifier – (obvykle) hash klíče
  - slouží aplikacím pro rychlé hledání

Další používaná rozšíření

- Key Usage – bitová mapa o možnostech využití klíče
  - šifrování, podpis, revokace...
- Extended Key Usage – účel použití klíče
  - pouze koncové cerfitikáty
  - serverové TLS, pošta
  - časová razítka, podpis kódu
  - podpisy OCSP a další
- Name Constraints – omezení pro určité domény
  - pouze u certifikátů CA
- Certificate Policies – strojově čitelná pravidla vydání a použití
  - 2.23.140.1.2.1 = DV; 2.23.140.1.2.2 = OV; EV různě

Podpis certifikátu

- celý certifikát je podepsaný autoritou
- nejprve ID funkce a algoritmu
- sha256WithRSAEncryption
- poté následuje samotný podpis

```
71:84:5e:01:a2:b3:e6:2f:c0:00:19:fa:49:46:42:7e:f9:6d:
c9:b9:b3:fa:f9:d3:1c:a6:5a:1f:20:f3:84:9a:8c:a0:c5:f2:
09:9d:89:b9:a5:b9:a0:a5:d8:3c:99:07:a8:d9:8d:87:32:c0:
0d:32:64:8e:f5:c1:c5:13:6f:09:7b:d7:69:4c:2d:62:bf:db:
8b:d5:86:fc:ad:8d:ab:45:31:49:0b:a7:29:62:82:b8:6f:68:
41:61:e8:bb:df:22:58:96:cb:f9:60:19:ec:18:3b:eb:33:e2:
c8:60:46:f6:21:94:58:d7:f6:50:02:77:27:bf:80:d4:11:7e:
dc:19:c3:08:23:76:9b:9c:60:c1:4d:22:b0:d2:c8:7b:2f:9b:
84:1f:73:f0:07:51:8c:2b:a4:71:84:f4:6f:83:56:39:a5:9b:
…
```

Revokace certifikátů

- chceme možnost revokovat před vypršením
- měníme CA, končí služba, zmizely klíče...
- čím delší platnost, tím větší riziko
- CA poskytují rozhraní k ověření platnosti
  - CRL – seznamy neplatných certifikátů
  - OCSP – razítka potvrzující platnost
- kořenový certifikát nelze revokovat

CRL (Certificate Revocation List)

- veřejné seznamy revokovaných klíčů
- stahuje se po HTTP
- odkaz v rozšíření CRL Distribution Points
- identifikátorem je sériové číslo certifikátu
- dále datum revokace a důvod
- vše podepsáno klíčem CA
- revokovaný cert musí být v CRL až do konce platnosti

```
$ openssl crl -inform DER -text -noout -in DSTROOTCAX3CRL.crl
```

Nevýhody CRL

- nutnost pravidelného stahování a obnovování
- poměrně velké objemy
- síťová zátěž na klienta i autoritu
  - problém na moblních datech
  - neškáluje to ani u autority
- čím víc certifikátů, tím delší seznamy
- klient musí mít komplexní knihovny a databázi
- prohlížeče od CRL upouštějí

OCSP (Online Certificate Status Protocol)

- jednoduchý protokol pro revokační dotazy
- definuje ho RFC 6960
- dotazy probíhají po HTTP
- klient se ptá OCSP responderu
- odpověď je krátká – týká se jednoho certifikátu
- tři možné stavy odpovědi
  - good
  - revoked
  - unknown
- zpráva je digitálně podepsaná autoritou
- stejnou nebo může CA delegovat na OCSP autoritu
  - ta může mít jiný klíč a nemůže vydávat certifikáty

Výhody a nevýhody OCSP

Výhody

- malý dotaz a odpověď
- odpověď má dobu platnosti
- dá se cachovat

Nevýhody

- klienti zasypávají autoritu dotazy
- autorita ví, kdo kdy kam přistupuje
- neškáluje to

Řešení: OCSP stapling

- díky podpisu je validační zprávu možné předat
- je platná sama o sobě
- stapling = přicvaknutí k certifikátu
- responderu se ptá server a odpověď přicvakne
- klient se nemusí ptát sám
- škáluje to, neunikají osobní údaje
- původní stapling neumí požádat o více razítek najednou
- problém pro mezilehlé – neumíme se serveru zeptat
- musíme zjišťovat revokace jinou cestou
- řeší RFC 6961 – umí poslat dotaz na víc certifikátů

Podpora staplingu v serverech a klientech

- Apache od 2.3.3
- nginx od 1.3.7
- Microsoft IIS od 2008
- Firefox od 3, od 26 stapling
- Internet Explorer od Vista
- Safari od OS X 10.7
- Opera od 8.0
- Exim v server i klient módu
- Chrome nepodporuje – šíří si seznamy sám

Praktická ukázka dotazu na OCSP

- potřebujeme koncový i mezilehlé – ověřuje se proti vystavovateli

```
$ openssl s_client -connect petrkrcmar.cz:443 -showcerts 2>&1 < /dev/null
```

- uložíme zvlášť do souborů chain.crt a petrkrcmar.crt
- v certifikátu najdeme OCSP URI

```
$ openssl x509 -in petrkrcmar.crt -noout -ocsp_uri
```

- položíme dotaz pomocí OpenSSL

```
$ openssl ocsp -no_nonce -issuer chain.crt -verify_other chain.crt \\
    -cert petrkrcmar.crt -url http://ocsp.int-x3.letsencrypt.org/ \\
    -header "HOST" ocsp.int-x3.letsencrypt.org [-text]
```

Praktická ukázka OCSP stapling

- stačí se připojit ke klientovi, který stapling umí

```
$ openssl s_client -connect petrkrcmar.cz:443 -status < /dev/null
```

- na začátku komunikace je info o OCSP

```
CONNECTED(00000003)
OCSP response:
====================================
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: C = US, O = Let's Encrypt, CN = Let's Encrypt Authority X3
    Produced At: Apr 28 09:12:00 2016 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: 7EE66AE7729AB3FCF8A220646C16A12D6071085D
      Issuer Key Hash: A84A6A63047DDDBAE6D139B7A64565EFF3A8ECA1
      Serial Number: 0304B9F9427D8E6EDFE48B0343C7EFB2653C
    Cert Status: good
    This Update: Apr 28 09:00:00 2016 GMT
    Next Update: May  5 09:00:00 2016 GMT
```

Otázky?

Petr Krčmář
petr.krcmar@iinfo.cz