

# Principy a správa DNS

Ondřej Caletka

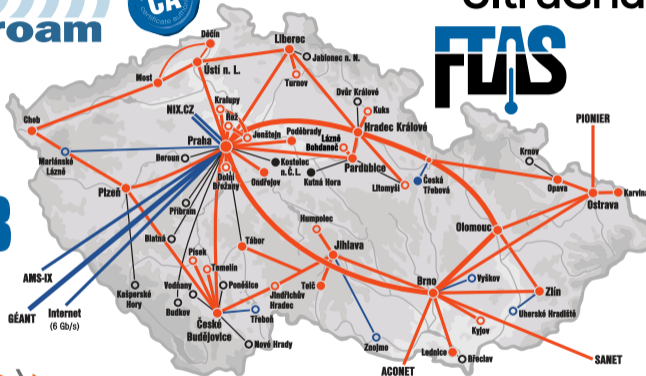


15. června 2017



Uvedené dílo podléhá licenci Creative Commons Uvedte autora 3.0 Česko.

# O sdružení CESNET

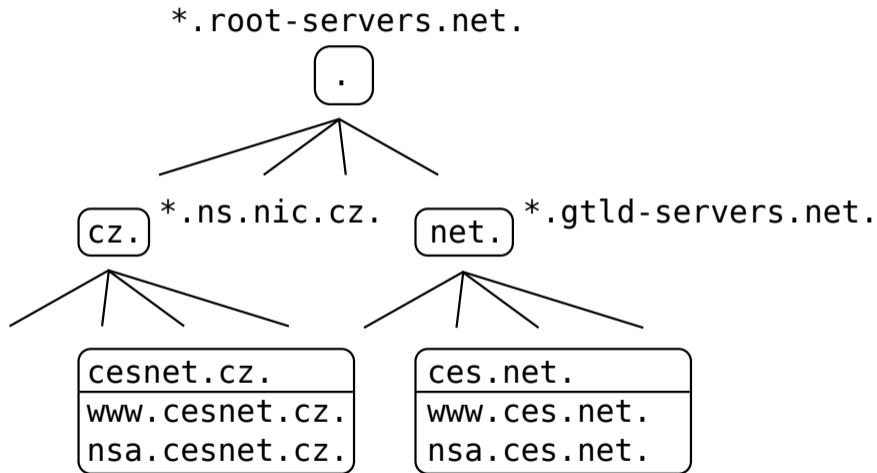


Orange line	$\geq 100$ Gb/s	Blue line	100 Gb/s
Red line	$\geq 10$ Gb/s	Blue line	10 Gb/s
Black dot	uzel (PoP)	Blue circle	1-2,5 Gb/s
Blue circle	uživatel (user)	Grey line	< 1 Gb/s

- 1 O službě DNS
- 2 Lab 1: Instalace rekurzivního resolveru
- 3 Autoritativní servery
- 4 Dynamické DNS
- 5 Lab 2: Zprovoznění autoritativního DNS serveru
- 6 DNSSEC na autoritativním serveru
- 7 Lab 3: DNSSEC na autoritativním serveru
- 8 Útoky zneužívající DNS
- 9 Budoucnost, tipy a triky

- navrženo v roce 1982 jako náhrada HOSTS.TXT
- hierarchická distribuovaná databáze
- důraz na dostupnost, namísto rychlých změn a plné konzistence
- binární protokol používající UDP a TCP spojení na známém portu 53

# Hierarchická struktura DNS zón



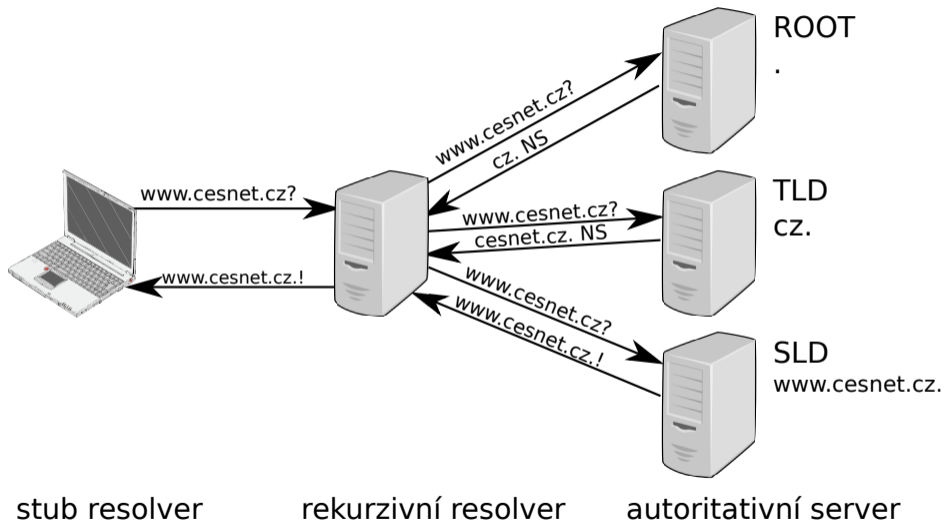
**zóna** část globální databáze, samostatně spravovaná  
např.: zóna cz. spravovaná sdružením CZ.NIC

**autoritativní server** server poskytující odpovědi ze zón, které drží  
např.: a.ns.nic.cz.

**rekurzivní server/resolver** server, který dokáže postupnými dotazy  
zjistit odpověď na libovolný DNS dotaz  
např.: Google Public DNS 8.8.8.8

**stub resolver** knihovná funkce, tvoří rozhraní mezi aplikací a resolverem  
např.: GNU C library (glibc)

# Tři druhy DNS nodů



- binární formát
- společné záhlaví
  - ID transakce
  - stavový kód
  - příznaky (později rozšířeny o DNSSEC)
    - AA Authoritative Answer
    - RD Recursion Desired
    - RA Recursion Available
    - TC TrunCated message
- čtyři sekce s *resource records*
  - QUERY dotaz
  - ANSWER konečná odpověď
  - AUTHORITY odkaz (referral)
  - ADDITIONAL doplňující informace



# DNS Resource Record

- udržován v cache po dobu TTL
- názvy domén jako spojový seznam *labels*
- komprese opakujících se názvů

www.cesnet.cz. 3600 IN A 195.113.144.230

```
- Questions: 1
- Answer RRs: 2
- Authority RRs: 4
- Additional RRs: 11
v Queries
  > www.cesnet.cz: type A, class IN
v Answers
  > www.cesnet.cz: type A, class IN, addr 195.113.144.230
  > www.cesnet.cz: type RRSIG, class IN
> Authoritative nameservers
> Additional records
.....
0000 78 2b cb aa 53 cf 00 15 2c 31 b8 00 08 00 45 00 x+..S... ,1....E.
0010 05 10 8b 97 00 00 37 11 75 42 9e c4 95 09 c3 71 .....7. uB.....q
0020 86 c4 00 35 96 7c 04 fc 33 a5 d1 5e 84 10 00 01 ...5.|.. 3.^....
0030 00 02 00 04 00 0b 03 77 77 77 06 63 65 73 6e 65 .....w ww.cesne
0040 74 02 63 7a 00 00 01 00 01 c0 0c 00 01 00 01 00 t.cz.... .....
0050 00 0e 10 00 04 c3 71 90 e6 c0 0c 00 2e 00 01 00 .....q. ....
0060 00 0e 10 00 0d 00 01 0e 02 00 00 0e 10 52 ca b2 .....s
```



# Typy záznamů

**A** IPv4 adresa

**AAAA** IPv6 adresa

**PTR** reverzní záznam

adresa se převrátí a připojí pod strom `in-addr.arpa.`, nebo `ip6.arpa.`

**MX** Mail eXchange - SMTP server

**CNAME** Canonical Name - alias

nelze kombinovat s jiným typem RR pro stejné jméno;  
neměl by se řetězit

**SRV** hledání služeb (SIP, XMPP, atd.)

**SSHFP** SSH finger print

**TLSA** TLS certifikát (DANE)



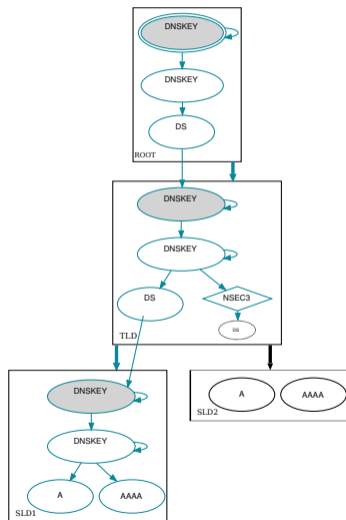
- historický limit UDP DNS paketu 512 B
- později přidána rozšiřující hlavička jako záznam typu EDNS0 v poli ADDITIONAL
  - inzeruje podporovanou délku UDP paketu (např. 4096 B)
  - další pole s příznaky
    - DO DNSSEC OK
- větší UDP zpráva šetří používání TCP, zhoršuje ale následky zesilujících útoků
- EDNS0 může další volby:
  - informaci o klientské podsíti (problém CDN vs. 8.8.8.8)
  - informaci o podporovaných DNSSEC algoritmech

- zaručení autenticity DNS zpráv
- využívá princip elektronického podpisu
- bez PKI, hierarchická důvěra
- validace nejen rekurzivními resolversy
- k validaci je potřeba nakonfigurovat pevný bod důvěry (*Trust Anchor*; obvykle otisk klíče kořenové zóny)
- nové příznaky:

AD Authenticated Data

CD Checking Disabled

DO DNSSEC OK



# Výsledek DNSSEC validace

**secure** validátor úspěšně sestavil řetěz důvěry od TA až po koncovou entitu a všechny podpisy souhlasí

**insecure** validátor úspěšně sestavil řetěz důvěry od TA až k podepsanému důkazu neexistence bezpečné delegace

**bogus** řetěz důvěry se nepodařilo sestavit (nesprávné nebo chybějící klíče, expirované podpisy,...)

**indeterminate** pro daný podstrom není nakonfigurován TA

**Poznámka:** Používá-li bezpečná delegace algoritmus, kterému validátor nerozumí, má se za to, že jde o důkaz neexistence bezpečné delegace.

# Lab 1: Instalace rekurzivního resolveru

# Lab 1: Instalace rekurzivního resolveru

- připojení k virtuálním serverům
- práce s příkazy `dig` a `host`
- instalace a konfigurace serverů Unbound a/nebo BIND

# Autoritativní servery



# Zónový soubor

- normalizovaná textová podoba jedné DNS zóny
- začíná záznamem typu SOA (Start of Authority)
  - jméno primárního serveru
  - e-mail hostmastera
  - sériové číslo
  - časovací parametry
- apex zóny (@)
  - obsahuje SOA, NS, apod. pro doménu bez prefixu
- tečka na konci jména určuje FQDN oproti relativnímu jménu
- řídicí direktivy
  - `$ORIGIN` doména připojená za relativní názvy
  - `$INCLUDE` vložení dalšího souboru
  - `$TTL` výchozí hodnota TTL

# Zónový soubor – příklad

```
$TTL      3600
@         IN      SOA  nsa.cesnet.cz. ( ;primary nameserver
                        hostmaster.cesnet.cz. ;admin e-mail
                        2012072500 ; serial
                        28800      ; refresh  ( 8 hod)
                        7200       ; retry   ( 2 hod)
                        1814400    ; expire  (21 dni)
                        900        ; neg. TTL (15 min)

                        IN  NS     nsa.cesnet.cz.
                        IN  NS     nsa.ces.net.
                        IN  NS     decsys.vsb.cz.

;
localhost IN  A     127.0.0.1
```

# Delegace a subdelegace

- způsob, jak je sestaven DNS strom
- nadřazená zóna obsahuje NS záznam s adresou serveru s zónou nižší úrovně  
např.: `cz. IN NS a.ns.nic.cz.`
- pokud server pro zónu leží uvnitř stejné zóny, je třeba navíc *glue* záznam  
např.: `a.ns.nic.cz. IN A 194.0.12.1`
- tyto informace se použijí pouze pro prvotní nasměrování (*priming*)  
po spojení s delegovaným serverem jsou v cache přepsány informacemi z apexu cílové zóny
- záznamy patřící do subdelegace DNS server ignoruje

# Subdelegace – příklad

```
$ORIGIN example.com.  
$TTL      3600  
@         IN  SOA  ...  
         IN  NS   ns1           ; NS v apexu není delegace  
ns1       IN  A    192.0.2.1  
  
sub       IN  NS   ns.sub        ; toto je delegace  
         IN  NS   server.nekde.cz.  
ns.sub    IN  A    192.0.2.2     ; glue záznam - nutný  
server.nekde.cz. IN A 192.0.2.3 ; nonsens - out-of-zone data  
server.sub IN A    192.0.2.4     ; nonsens - data v delegované  
                                     ; zóně, která nejsou glue
```

# Reverzní delegace

adresa se převrátí (IPv4 po oktetech, IPv6 po nibblech) a připojí pod strom in-addr.arpa., nebo ip6.arpa.

## IPv4

```
server.example.com.      IN A    192.0.2.1
1.2.0.192.in-addr.arpa. IN PTR  server.example.com.
```

## IPv6

```
server.example.com.      IN AAAA 2001:db8:123:456::1
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.\
6.5.4.0.3.2.1.0.8.b.d.0.1.0.0.2.ip6.arpa.
                           IN PTR  server.example.com.
```

# Reverzní classless delegace

*Problém:* rozsahy IPv4 adres jsou menší, než celá třída.

**zóna 2.0.192.in-addr.arpa.**

```
128/25  IN NS server.example.com.  
        IN NS secondary.example.com.  
128     IN CNAME 128.128/25  
129     IN CNAME 129.128/25  
...  
255     IN CNAME 255.128/25
```

**zóna 128/25.2.0.192.in-addr.arpa**

```
129     IN PTR server.example.com
```

- synchronizace autoritativních serverů
- slave servery periodicky dotazují SOA master serveru
- došlo-li ke zvýšení sériového čísla, požádají pomocí TCP o záznam typu AXFR, nebo IXFR
- master server odpoví kompletním obsahem zóny (AXFR), nebo změnou proti předchozímu sériovému číslu (IXFR)
- není-li master dlouho dostupný, zóna expiruje
- master může upozornit slave servery zprávou NOTIFY

# Zabezpečení přenosu pomocí TSIG

- zabezpečení DNS dotazu elektronickým podpisem
- využívá sdílené tajemství a algoritmus HMAC
- lze použít např. místo omezování IP adres pro zónové přenosy
- chrání před softwarovými chybami (pád DNS serveru uprostřed přenosu)

## Vygenerování sdíleného tajemství

```
$ openssl rand -base64 32  
UKoj75Qy5B0Gb0KxRDJhtKRQkdYXmrsIPcdy2nBchJI=
```



# Časování a synchronizace

- odpovědi serverů kešovány po TTL daného záznamu
- negativní odpovědi kešovány podle hodnoty SOA minimum
- nesynchronnost serverů vede ke *split-brain*:  
o odpovědi rozhoduje náhoda

## Za jak dlouho se změna nejpozději projeví?

	s NOTIFY	bez NOTIFY
nový změna	SOA minimum TTL starého	SOA minimum + SOA refresh TTL starého + SOA refresh

# Dynamické DNS

- tradiční DNS servery načítají zónový soubor
- změna dat vyžaduje změnu zónového souboru a reload serveru
- dynamické DNS je rozšíření DNS protokolu o možnost aktualizace dat
- server změny aplikuje automaticky a zvyšuje sériové číslo
- po zapnutí DDNS není již nadále možné editovat zónové soubory

# Zprovoznění DDNS

- 1 nastavíme TSIG klíč
- 2 povolíme dynamické aktualizace
- 3 aktualizujeme utilitou nsupdate

## Příklad nsupdate

```
> server nXX.clones.cesnet.cz  
> update delete test.example.com.  
> update add test.example.com. 60 IN TXT "test"  
> send
```

- nevýhodou DDNS je ztráta formátování a komentářů ve zdrojových souborech
- není možné kombinovat DDNS a editaci zónového souboru na jedné zóně
- možným řešením je utilita `nsdiff`. Ta porovná starou zónu s novou a vygeneruje skript pro `nsupdate`, který změny aplikuje.

```
$ nsdiff example.com example.com.zone | nsupdate
```

# Lab 2: Zprovoznění autoritativního DNS serveru

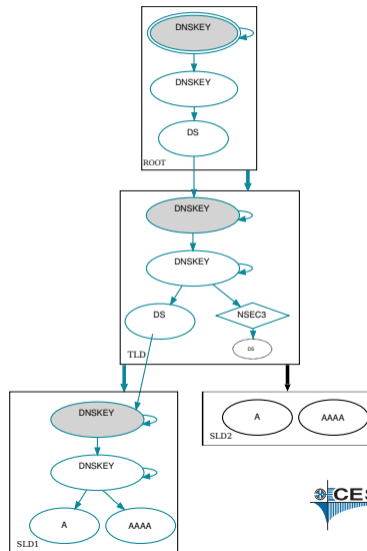
# Lab 2: Zprovoznění autoritativního DNS serveru

- instalace a konfigurace serverů BIND, Knot, NSD
- vytvoření delegace z nadřazené zóny
- master-slave replikace

# DNSSEC na autoritativním serveru



- rozšíření zajišťující autenticitu DNS odpovědí
- navržen pro podporu off-line podepisování
- nové typy záznamů:
  - RRSIG podpis RR
  - DNSKEY veřejný klíč
  - NSEC next secure
- nové příznaky:
  - AD Authenticated Data
  - CD Checking Disabled
  - DO DNSSEC OK



- ke každému RR je vygenerován podpis RRSIG
- platnost podpisu je časově omezena (typ. 14 dnů)
- veřejný klíč, pomocí kterého je možné RRSIG ověřit, je uložen v záznamu DNSKEY v apexu zóny
- nadřazená zóna přidá k delegaci DS záznam s otiskem veřejného klíče zóny
- otisk klíče kořenové zóny získá validátor jinou cestou

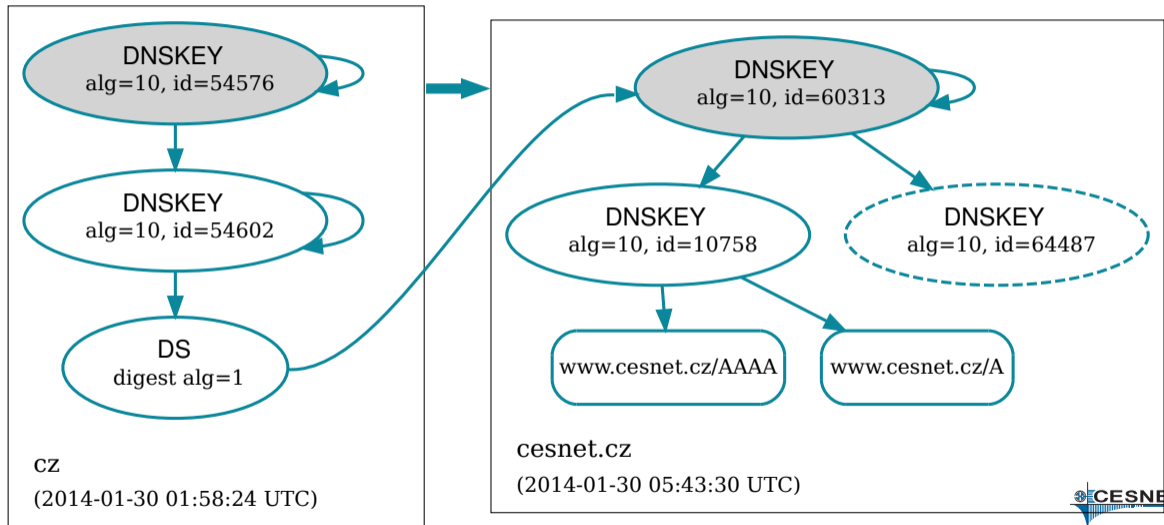
# Klíče a podpisy – příklad

```
cesnet.cz. IN DNSKEY 257 3 10 (
    AwEAAAdJGR0Kw52qqSwZsxLRfIu
    ...
    cC3rtivHNCzPX/xafcBBalwZE=
    ) ; KSK; alg = RSASHA512; key id = 60313

cesnet.cz. IN RRSIG DNSKEY 10 2 3600 (
    20140907110144 20140808092435
    60313 cesnet.cz.
    08MIEeUT/reCCKZd0W57hDdP4b1mnd+zE
    ...
    UdHhvKkTyI0KPJx2BQh+cV+pub+6rEFycw== )
```

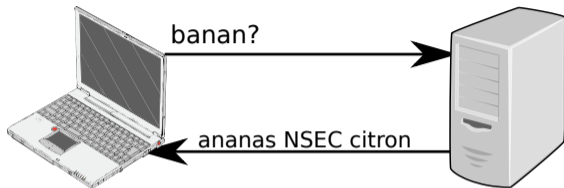
- snaha učinit podpisy dostatečně krátké vede k používání ne příliš bezpečných klíčů (1024bit RSA)
- takové klíče by se měly často měnit
- komunikace s nadřazenou zónou nemusí být jednoduchá a je snaha se jí vyhnout
- řešením je dvojice klíčů: *Zone Signing Key* a *Key Signing Key*
- KSK je silný klíč, který podepisuje jen DNSKEY záznam; jeho otisk je v nadřazené zóně
- ZSK je slabý klíč, který podepisuje všechny RR; při jeho výměně se jen přepodepíše DNSKEY pomocí KSK

# KSK a ZSK – příklad



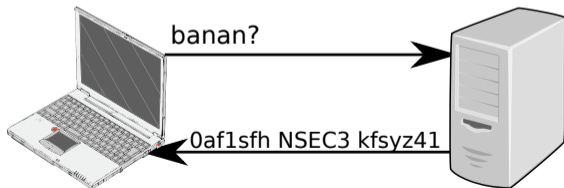
- řeší problém věrohodného popření existence záznamu (bez generování podpisů v reálném čase)
- při dotazu na neexistující záznam je vrácen podepsaný záznam NSEC pokrývající lexikální prostor mezi předchozím a následujícím existujícím záznamem
- NSEC záznamy tvoří spojový seznam, pomocí kterého je možné projít všechny záznamy v zóně
- při online signingu lze dynamicky generovat nejmenší možné NSEC záznamy (NSEC white lies)
- NSEC3 používá solené jednosměrné hashe, které procházení znesnadňují

# NSEC a NSEC3



```
@      IN NSEC ananas  
ananas IN NSEC citron  
citron IN NSEC @
```

```
hash(@)      = 0af1sfh  
hash(ananas) = z781sfa  
hash(citron) = kfsyz41  
hash(banan)  = 27acj54
```



```
0af1sfh IN NSEC3 kfsyz41  
kfsyz41 IN NSEC3 z781sfa  
z781sfa IN NSEC3 0af1sfh
```

# Wildcard záznamy a DNSSEC

- server k syntetické odpovědi přiloží RRSIG k wildcardu s nižším počtem v poli labels
- navíc musí přiložit NSEC důkaz, že neexistuje specifitější záznam
- spouta *buggy* implementací
  - djbdns vrací špatné NSEC záznamy
  - NSD dříve nevracelo redundantní NSEC záznamy, které potřeboval BIND
  - rekurzivní BIND <9.9 neposílá NSEC záznamy dalšímu, problém s řetězením



# Možnosti nasazení DNSSEC

- 1 ruční podepisování utilitou `signzone`
- 2 DNSSEC blackbox (komerční, OpenDNSSEC)
- 3 on-line podepisování v nameserveru (vyžaduje obvykle DDNS)
  - BIND
  - Knot DNS
  - PowerDNS

# Základem je dostatek entropie

- generování klíčů i podpisů potřebuje náhodná čísla
- standardně se používá `/dev/random`
- není-li v systému dostatečná entropie, krypto funkce vytuhávají
- použití `/dev/urandom` *nemusí být* bezpečné při nedostatku entropie
- *jiné* řešení: instalace `haveged`

```
# apt-get install haveged
```

LCE: Don't play dice with random numbers

# Vygenerování klíčů

```
# dnssec-keygen -a RSASHA256 -b 2048 -f KSK example.com
Generating key pair.....+++ .....+++
Kexample.com.+008+32797
# dnssec-keygen -a RSASHA256 -b 1024 example.com
Generating key pair...+++++..... ..+++++
Kexample.com.+008+46884
```

Vzniknou soubory `.key` a `.private`. Obsah prvního vložíme do zóny.

```
# dnssec-signzone -3 cafe -x -N unixtime example.com
Verifying the zone using the following algorithms: RSASHA256.
Zone fully signed:
Algorithm: RSASHA256: KSKs: 1 active, 0 stand-by, 0 revoked
                    ZSKs: 1 active, 0 stand-by, 0 revoked
example.com.signed
```

Vznikne soubor `example.com.signed`. Toto je třeba opakovat po každé změně dat zóny, stejně jako po určité době (životnost podpisů je standardně 30 dnů)

## Metoda předpublikace

- 1 vystavíme nový klíč
- 2 počkáme, až se rozšíří
- 3 začneme podepisovat novým klíčem
- 4 počkáme až zmizí staré podpisy
- 5 vymažeme starý klíč

- ✓ vždy pouze jedna sada podpisů
- ✗ zdlouhavé
- ✗ při použití na KSK nutná dvojí komunikace s nadřazenou zónou

## Metoda dvojího podpisu

- 1 vystavíme nový klíč, podepíšeme oběma
- 2 počkáme, až se rozšíří
- 3 vyměníme *DS záznam v nadřazené zóně*
- 4 počkáme, až se změna *DS rozšíří*
- 5 odstraníme starý klíč a staré podpisy

- ✓ rychlejší
- ✓ jediná komunikace s nadřazenou zónou
- ✗ při použití na ZSK by objem dat enormě narostl

- kompletní nástroj na nasazení DNSSEC
- nepodepsaná zóna na vstupu – podepsaná zóna na výstupu
- k uložení klíčů používá HSM s PKCS#11 rozhraním, případně SoftHSM
- politika *Key and Signature Policy* určuje všechny volitelné parametry DNSSECu
- může pracovat jako mezistupeň v zónovém přenosu

- dostupné v BIND, Knot, PowerDNS
- obvykle vyžaduje DDNS přístup
- klíče je často nutné generovat ručně
- dva přístupy:
  - předgenerování podepsané zóny – BIND, Knot
  - generování podpisů v reálném čase – PowerDNS, Knot



# In-line signing v BIND 9.9

- automaticky pravidelně podepisuje zónu
- nemění původní zónové soubory
- vyžaduje ruční generování klíčů

Na nás tedy zbývá:

- 1 vygenerovat klíč
- 2 upravit konfiguraci
- 3 publikovat DS záznam v nadřazené zóně

<https://www.root.cz/clanky/dnssec-s-bind-9-9-snadno-a-rychle/>

# Lab 3: DNSSEC na autoritativním serveru

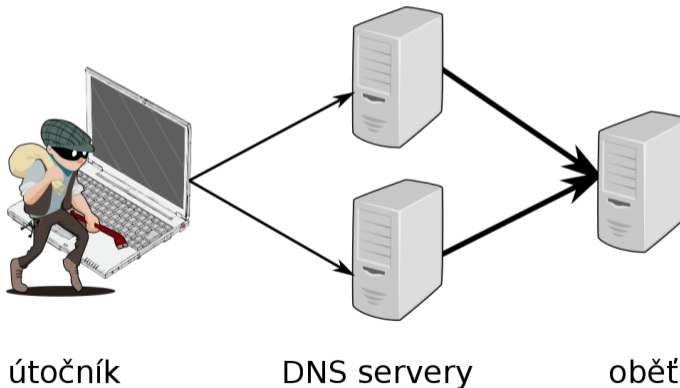
# Lab 3: DNSSEC na autoritativním serveru

- vygenerování klíčů
- podepsání zóny nástroji `bind9utils`
- výměna klíčů

# Útoky zneužívající DNS

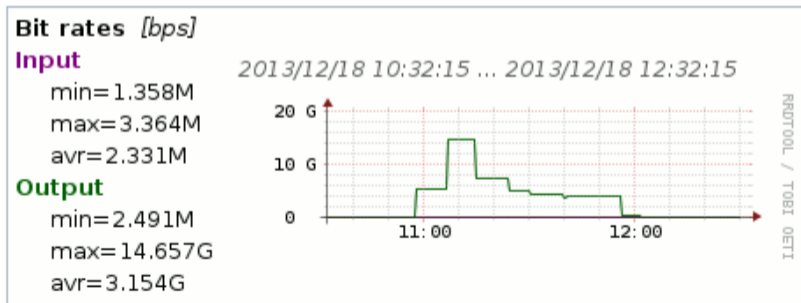
# Útoky na/pomocí DNS

- odepření služby zahlcením
- odepření služby vyčerpáním prostředků
- zesilující útok odrazem od DNS serverů



# Odepření služby zahlcením

- incident 18. 12. 2013 11:00 – 12:00 CET
- zahlcení hlavního DNS resolveru UDP pakety na náhodná čísla portů, *obsahující 128 × 0x00*
- provoz přicházel ze všech zahraničních linek z náhodných adres
- možné protiopatření: ACL na hraničních routerech



# Potírání zesilujících útoků

- implementujte BCP 38 (a nuťte ostatní)
- neotvírejte rekurzivní servery do světa  
*a zkontrolujte taky NTP servery a zařízení se SNMP ☺*
- na autoritativních serverech zapněte RRL

## Response Rate Limiting

Obecná technika limitování odpovědí autoritativních serverů na opakující se dotazy ze stejné adresy. Implementováno nativně v Knot DNS, NSD, i BIND 9.9.

# Útok náhodnými subdoménami

- mladá forma útoku (2014) zneužívající otevřené rekurzivní resolversy
- pro rekurzivní resolver připomíná Slowloris útok
- postihuje zároveň rekurzivní i autoritativní servery
- útočící botnet pokládá dotazy ve stylu `<random string>.www.obet.com`
- dotaz je vždy přeposlán autoritativnímu serveru
- autoritativní server se buď pod nápořem zhroutí, nebo zasáhne rate limiting
- rekurzivní server čeká na odpověď a zkouší dotazy opakovat

<https://www.root.cz/clanky/utok-na-dns-nahodnymi-dotazy/>



# Náhodné subdomény – důsledky a obrana

## důsledky

- zahlcení serverů dotazy
- DoS rekurzivních resolverů, např. BIND:
  - maximum 1000 současně probíhajících rekurzí
  - každá rekurze používá jeden file descriptor
  - pro víc než ~4000 rekurzí přestává být spolehlivý

## obrana

- definování prázdných SLD zón obětí na rekurzoru
  - riziko zablokování významných domén jako `in-addr.arpa`, nebo `co.uk`
- volba `ratelimit` v Unbound, `fetches-per-server` v BIND

# Omezení velikosti UDP odpovědi

- rozšíření EDNS0 zvětšuje délku UDP zpráv nad 512 B  
*obvykle na 4096 B*
- omezením velikosti k  $\sim 1$  kB snížíme účinnost zesilujícího útoku
- také se tím zlepší situace resolverům s nefunkčním *Path MTU Discovery*
- příliš nízká hodnota může naopak rozbít resolversy bez TCP konektivity
  - obzvláště při použití DNSSEC
  - takto postižených uživatelů je  $\sim 2$  % (měření Geoffa Hustona)

# RRL v linuxovém firewallu

- pouze jako dočasné řešení před nasazením RRL
- modul hashlimit pro netfilter
- vlastní modul xt\_dns pro klasifikaci typu DNS provozu

```
Domain Name System (query)
├── [Response In: 2]
├── Transaction ID: 0x3aab
├── > Flags: 0x0100 (Standard query)
├── Questions: 1
├── Answer RRs: 0
├── Authority RRs: 0
├── Additional RRs: 0
├── Queries
│   └── nebezi.cz: type ANY, class IN
│       ├── Name: nebezi.cz
│       ├── Type: ANY (Request for all records)
│       └── Class: IN (0x0001)
└── ...
0000  00 00 00 00 00 00 00 00 00 00 00 00 00 08 00 45 00  .....E.
0010  00 37 88 40 00 00 40 11  f4 73 7f 00 00 01 7f 00  .7.@..@. .s.....
0020  00 01 cc 4f 00 35 00 23  fe 36 3a ab 01 00 00 01  ...0.5.# .6:.....
0030  00 00 00 00 00 00 06 6e  65 62 65 7a 69 02 63 7a  .....n ebezi.cz
0040  00 00 ff 00 01  .....
```

<https://www.root.cz/clanky/zabezpecte-svuj-dns-server/>

# Únos domény

- týká se zejména vysoce hodnotných domén
- nejčastěji předelegování na jinou sadu serverů
- extrémně dlouhá TTL (týden)
- útočné vektory:
  - zastaralé delegace v TLD
  - únos IP adresy autoritativního serveru
  - **hacknutí registrátora**
- obrana:
  - DNSSEC
  - TSIG
  - registry-level lock (<https://www.domenovyprohlizec.cz>)

<https://www.root.cz/clanky/za-vypadek-znameho-blacklistu-mohla-unesena-domena/>



# Budoucnost, tipy a triky

# Proč nepoužívat obskurní DNS servery

```
$ host www.skvelabanka.cz  
www.skvelabanka.cz has address 192.0.2.7  
Host www.skvelabanka.cz not found: 3(NXDOMAIN)
```

```
$ host www.skvelabanka.cz  
Host www.skvelabanka.cz not found: 3(NXDOMAIN)
```

- programátor nepředpokládal, že se někdo zeptá na MX záznam pro `www.skvelabanka.cz`
- jeho implementace na takový dotaz vracela NXDOMAIN s TTL = 1 hodina
- BIND takovou odpověď nakešoval a po dobu TTL nevracel žádná data pro `www.skvelabanka.cz`



# On-line kontroly

**ZONEMASTER**  
by *net* and *afnic*

Domain check Pre-delegated domain FAQ Non-Javascript Interface

Domain name  
ces.net

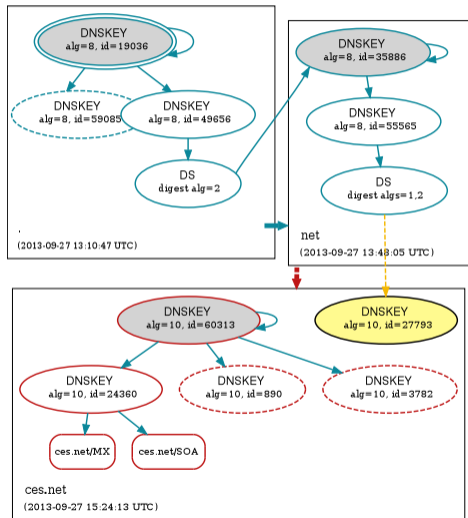
Advanced options

**Test # 166751**  
Executed at 2016-08-29T17:00+0200  
Link  
<https://www.zonemaster.fr/test/9df65d640f1bd79>

Basic Advanced Export History

- ✓ SYSTEM
- ✓ BASIC
- ✓ ADDRESS
- △ CONNECTIVITY
- ✓ CONSISTENCY
- ✓ DNSSEC
- ✓ DELEGATION
- ✓ NAMESERVER
- ✓ SYNTAX
- ✓ ZONE

Zonemaster Test Engine Version:v1.0.14, IP address: 2001:718:110:134:156  
contact@zonemaster.net Github



# Zjištění IP adresy resolveru

## Zjištění vlastní adresy

```
$ dig +short o-o.myaddr.l.google.com txt \
  @ns1.google.com
"195.113.134.196"
```

## Zjištění adresy DNS resolveru

```
$ dig +short o-o.myaddr.l.google.com txt
"195.113.187.90"
```

## Podpora EDNS0 client subnet

```
$ dig +short o-o.myaddr.l.google.com txt @8.8.8.8
"74.125.47.19"
"edns0-client-subnet 195.113.134.0/24"
```



# SSHFP záznamy

umístění otisku serverového klíče do DNS

## Vygenerování klíče

```
$ ssh-keygen -r server
server IN SSHFP 1 1 b2...16
server IN SSHFP 1 2 e9...a307881a26da5961f41ef41ccc
server IN SSHFP 2 1 6c...57
server IN SSHFP 2 2 1e...44963ffbf82b1c028d365b859e
server IN SSHFP 3 1 3f...a3
server IN SSHFP 3 2 a9...9d7dd752bea56ff505281c7ed1
```

## Validace

```
$ echo "VerifyHostKeyDNS yes" >> ~/.ssh/config
```

<https://www.root.cz/clanky/dnssec-jako-bezpecne-uloziste-ssh-klicu/>



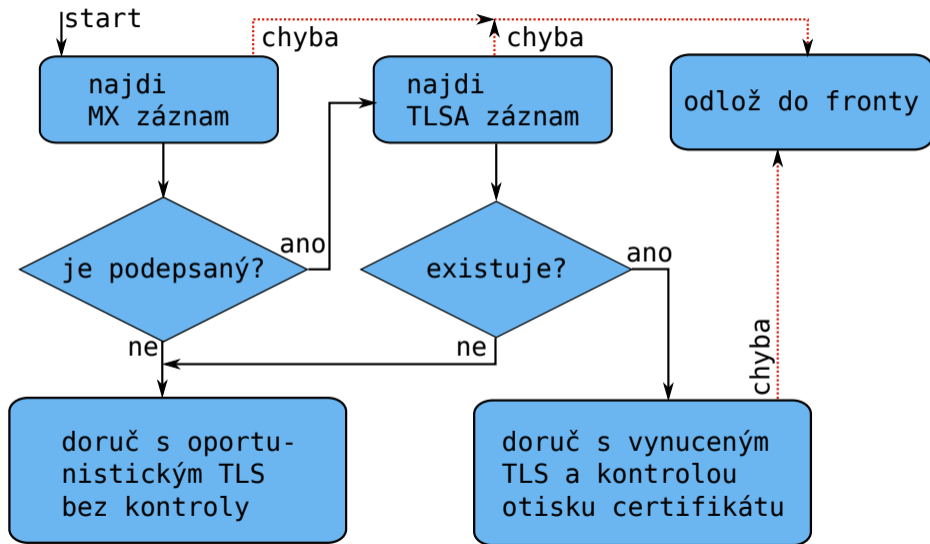
# TLSA záznamy (DANE)

- generujte pomocí utility swede
- zvolte usage podle vašeho vztahu s CA:
  - 0/2 připíchnutí/vložení nové CA
  - 1/3 připíchnutí/vložení nového koncového certifikátu
- **dodržujte správnou proceduru výměny otisků**
- **informujte všechny, kdo certifikáty vyměňují**



<https://www.root.cz/clanky/pripichnete-si-ssl-certifikat-k-domene/>

# Použití DANE pro SMTP



# Opt-in for security

- TLSA záznamem deklaruujeme, že poštu přijímáme pouze šifrovaně
- validující klienti případný *downgrade* útok odhalí a zprávu nedoručí
  - Postfix od 2.11 (leden 2014)
  - Exim – ve vývoji
  - OpenSMTPd – ve vývoji
- na rozdíl od webu na SMTP serverech není problém s funkčností DNSSEC validace
- bezpečné spojení s validujícím DNS serverem **je nutné** (ideálně Unbound na localhost)
- doručování na adresy bez DNSSECu nebo bez TLSA záznamu funguje jako doposud

# Dynamické IPv6 záznamy

- běžná praxe v IPv4: vygenerování záznamů pro každou IP adresu
- pro IPv6 nemožné, zóna pro /64 zabere stovky EiB ( $2^{60}$ )
- řešením je dynamické generování, podporované v Knot DNS 1.5+
- vyžaduje podporu ve všech autoritativních serverech zóny

## Příklad

```
example.cz {
  file "/etc/knot/empty.zone";
  query_module {
    synth_record "forward dyn- 60 2001:db8:1::/64";
    synth_record "forward dyn- 60 192.0.2.0/24";
  }
}
```

# IDN záznamy - Punycode

- umožňuje používat národní abecedy
- kompatibilní software automaticky převádí do punycode formy
- převod provádí stavový automat:
  - opakovaně prochází bufferem zleva doprava
  - po každém průchodu zvýší kód vkládaného znaku o 1
  - po uběhnutí zadaného počtu kroků vloží na danou pozici bufferu znak s daným kódem

## Příklady Punycode

xn--esnet-gya	česnet
xn--esnet-gyab	čečsnet
xn--a-iga9gb	číča
xn--eda7db	číč

# Eliptické křivky v DNSSEC

- výrazně kratší klíče a podpisy při srovnatelné síle
- umožní opustit koncept KSK a ZSK klíčů
- problematická validace staršími resolversy

## DNSSEC u CloudFlare

- první z velkých hráčů, který implementoval DNSSEC
- eliptické křivky, online signing
- odpovědi menší než 512 B
- expanze žolíků před podpisem, NSEC white lies

# Kompatibilita validátoru s různými algoritmy

```
$ go run alg_rep.go -r adns1.cesnet.cz
Zone dnssec-test.org. Qtype DNSKEY Resolver [adns1.cesnet.cz]
  debug=false verbose=false Prime= V
DS      :  1  2  3  4 |  1  2  3  4
ALGS    :      NSEC   |      NSEC3
alg-1   :  -  -  -  - |  x  x  x  x => RSA-MD5 OBSOLETE
alg-3   :  V  V  -  - |  x  x  x  x => DSA/SHA1
alg-5   :  V  V  -  - |  x  x  x  x => RSA/SHA1
alg-6   :  x  x  x  x |  V  V  -  - => RSA-NSEC3-SHA1
alg-7   :  x  x  x  x |  V  V  -  - => DSA-NSEC3-SHA1
alg-8   :  V  V  -  - |  V  V  -  - => RSA-SHA256
alg-10  :  V  V  -  - |  V  V  -  - => RSA-SHA512
alg-12  :  -  -  -  - |  -  -  -  - => GOST-ECC
alg-13  :  -  -  -  - |  -  -  -  - => ECDSAP256SHA256
alg-14  :  -  -  -  - |  -  -  -  - => ECDSAP384SHA384
V == Validates  - == Answer  x == Alg Not specified
T == Timeout  S == ServFail  0 == Other Error
DS algs 1=SHA1 2=SHA2-256 3=GOST 4=SHA2-384
```

[https://github.com/ogud/DNSSEC\\_ALG\\_Check](https://github.com/ogud/DNSSEC_ALG_Check)





# Posílení ZSK klíče kořenové zóny

- kořenová zóna dlouho používala RSASHA256 s klíči 2048 a 1024 bitů
- 1024bitové RSA podle některých není dostatečně bezpečné
- slabý klíč kořenové zóny *kompromituje* celý DNSSEC
- klíč byl vyměněn 1. října 2016 za 2048bitový
- vzrostla velikost odpovědi (a bude hůře)

# Root DNSSEC KSK rollover

- podpis kořenové zóny 15. července 2010
- rolování kořenového klíče podle potřeby, nebo jednou za pět let
- vyžaduje aktualizaci *trust anchor* ve všech validátorech
- proběhne automaticky ve většině případů (RFC 5011)
- testovací prostředí na <http://go.icann.org/KSKtest>
- mohou nastat problémy s velikostí odpovědi kořenových serverů
- aktuální jízdní řád:
  - nový KSK publikován 3. února 2017
  - vložení do zóny 11. července 2017
  - ukončení používání starého KSK naplánováno na 11. října 2017

Geoff Huston: Scoring the DNS Root Server System  
Root.cz: Klíč kořenové zóny se letos změní: správci serverů by měli dávat pozor



## RFC 7344 – automatická údržba DS záznamů

- klient publikuje v zóně CDS nebo CDNSKEY záznamy
- nadřazená zóna si toho všimne a upraví DS záznamy
- není určeno pro bootstrapping DNSSECu

## RFC 7477 – synchronizace delegací

- klient publikuje v apexu zóny CSYNC záznam
- nadřazená zóna si toho všimne a upraví NS záznamy a glue záznamy

## RFC 7816 – minimalizace dotazů

- místo celého jména se k autoritativnímu serveru posílá jen část
- naráží na rozbité servery s *obsahem pod NXDOMAIN* (RFC 8020)
- implementováno v Unboundu 1.5.7

## RFC 7858 – DNS-over-TLS

- tunelování DNS provozu TLS na novém portu 853
- volitelná autentizace serveru

<https://www.root.cz/clanky/soukromi-v-dns-kratke-dotazy-a-sifrovana-komunikace/>

# Závěr

- virtuální servery budou smazány
- předem díky za zpětnou vazbu

Děkuji za pozornost

Ondřej Caletka

Ondrej.Caletka@cesnet.cz

<https://Ondrej.Caletka.cz>

