Principy a správa DNS - cvičení

Ondřej Caletka, CESNET

3. prosince 2015

Obsah

1 Zprovoznění virtuálních serverů

Pro účely cvičení jste dostali k dispozici přístup na dvojici virtuálních serverů. Hlavní server, který budete používat nebude-li určeno jinak, má jméno `nXX.nebula.cesnet.cz`, kde `XX` je číslo vám přiděleného virtuálního serveru.

Pomocný server má jméno `nXXa.nebula.cesnet.cz`. Proti hlavnímu jsou jeho prostředky omezené – má méně paměti a nemá k dispozici IPv4 adresu. Na serveru je předinstalován Knot DNS a vytvořena zóna `XX.acad.cz` na kterou vede DNSSEC-podepsaná delegace z veřejného doménového jména `acad.cz`. Pomocný server budete používat hlavně pro nastavení delegace při pokusech s autoritativním serverem na hlavním serveru, případně si na něm můžete vyzkoušet zónové přenosy.

K přihlášení na oba servery použijte SSH na účet `root` pomocí privátního klíče, který jste obdrželi.

```
$ chmod 600 id_rsa-root@nXX.nebula.cesnet.cz
$ ssh -i id_rsa-root@nXX.nebula.cesnet.cz root@nXX.nebula.cesnet.cz
```

Privátní klíč také můžete překopírovat na hlavní server a ten pak používat jako přestupní stanici k pomocnému serveru. To se hodí zejména v případě, kdy má váš počítač problém s připojením prostřednictvím IPv6.

```
$ scp -i id_rsa-root@nXX.nebula.cesnet.cz \
> id_rsa-root@nXX.nebula.cesnet.cz root@nXX.nebula.cesnet.cz:.ssh/id_rsa
```

1.1 DNSViz

Na hlavním serveru je předinstalována offline verze nástroje DNSviz, který graficky analyzuje vztahy mezi zónami a DNSSEC podpisy. Pro použití stačí zavolat skript

```
# dnsviz-analyze cesnet.cz
Analyzing cesnet.cz
Analyzing cz
Analyzing .
```

Výsledek analýzy je vystaven v kořeni předinstalovaného web serveru a lze jej tedy prohlížet na adrese `http://nXX.nebula.cesnet.cz/cesnet.cz.html`. Standardně analýza probíhá prostřednictvím systémového DNS resolveru. Pro přímou analýzu autoritativních serverů přidejte na konec příkazového řádku slovo `auth`.

2 Ovládání utilit `host` a `dig`

2.1 `host`

Utilita `host` poskytuje vysokoúrovňový přístup k DNS resolveru. K zadanému doménovému jménu zjistí A záznam, AAAA záznam, a MX záznam, k zadané IP adrese zjistí PTR záznam. Druhým volitelným argumentem je adresa DNS serveru, kterému má být dotaz položen:

```
$ host cesnet.cz
cesnet.cz has address 195.113.144.230
cesnet.cz has IPv6 address 2001:718:1:101::4
cesnet.cz mail is handled by 100 mail.cesnet.cz.
cesnet.cz mail is handled by 10 postino.cesnet.cz.
cesnet.cz mail is handled by 50 cartero.cesnet.cz.

$ host 195.113.144.230 google-public-dns-a.google.com
Using domain server:
Name: google-public-dns-a.google.com
Address: 2001:4860:4860::8888#53
Aliases:

230.144.113.195.in-addr.arpa domain name pointer www.cesnet.cz.
```

Utilita má dále několik užitečných přepínačů. Přepínač `-C` například umožňuje snadno zkontrolovat synchronnost všech DNS serverů pro danou doménu:

```
# host -C cesnet.cz
Nameserver 195.113.144.228:
     cesnet.cz has SOA record nsa.cesnet.cz. hostmaster.cesnet.cz. 2015061103…
Nameserver 2001:718:1:101::144:228:
     cesnet.cz has SOA record nsa.cesnet.cz. hostmaster.cesnet.cz. 2015061103…
Nameserver 195.113.144.205:
     cesnet.cz has SOA record nsa.cesnet.cz. hostmaster.cesnet.cz. 2015061103…
Nameserver 2001:718:1:1::144:205:
     cesnet.cz has SOA record nsa.cesnet.cz. hostmaster.cesnet.cz. 2015061103…
Nameserver 2001:718:1001:149::9:
     cesnet.cz has SOA record nsa.cesnet.cz. hostmaster.cesnet.cz. 2015061103…
Nameserver 158.196.149.9:
     cesnet.cz has SOA record nsa.cesnet.cz. hostmaster.cesnet.cz. 2015061103…
```

2.2 `dig`

Utilita `dig` umožňuje přístup k DNS protokolu na nejnižší úrovni. V zásadě jen překládá binární DNS zprávy do standardizované textové podoby a zpět. Výstup příkazu je rozdělen do sekcí `QUESTION`, `ANSWER`, `AUTHORITY` a `ADDITIONAL` stejně jako DNS paket.

```
# dig cesnet.cz mx @8.8.8.8

; <<>> DiG 9.9.5-9-Debian <<>> cesnet.cz mx @8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 47330
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
```

```
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;cesnet.cz.                     IN      MX

;; ANSWER SECTION:
cesnet.cz.              3599    IN      MX      10 postino.cesnet.cz.
cesnet.cz.              3599    IN      MX      50 cartero.cesnet.cz.
cesnet.cz.              3599    IN      MX      100 mail.cesnet.cz.

;; Query time: 38 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Thu Jun 11 09:27:03 CEST 2015
;; MSG SIZE  rcvd: 107
```

Přepínač `+multi` přepne výstup do víceřádkového režimu s komentáři:

```
# dig +multi XX.acad.cz soa

; <<>> DiG 9.9.5-9-Debian <<>> +multi XX.acad.cz soa
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29622
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;XX.acad.cz.            IN SOA

;; ANSWER SECTION:
XX.acad.cz.            60 IN SOA nXXa.nebula.cesnet.cz. root.localhost. (
                                 2          ; serial
                                 300        ; refresh (5 minutes)
                                 150        ; retry (2 minutes 30 seconds)
                                 600        ; expire (10 minutes)
                                 60         ; minimum (1 minute)
                                 )

;; Query time: 18 msec
;; SERVER: 195.113.144.194#53(195.113.144.194)
;; WHEN: Thu Jun 11 13:10:55 CEST 2015
;; MSG SIZE  rcvd: 110
```

Příkaz `dig` je také možné použít na zjištění skutečného jména DNS serveru, případně i verze (hodí se pro anycast), posláním dotazu ve speciální třídě Chaos:

```
# dig hostname.bind ch txt @ns.cesnet.cz +short
"adns1.cesnet.cz"
```

Další příklady zajímavých služeb:

```
# dig +short o-o.myaddr.l.google.com txt
"195.113.187.254"
# dig +short porttest.dns-oarc.net txt
porttest.y.x.w.v.u.t.s.r.q.p.o.n.m.l.k.j.i.h.g.f.e.d.c.b.a.pt.dns-oarc.net.
"195.113.187.254 is GREAT: 72 queries in 4.4 seconds from 72 ports with std dev 19386"
# dig +short rs.dns-oarc.net txt
rst.x4050.rs.dns-oarc.net.
rst.x4058.x4050.rs.dns-oarc.net.
rst.x4064.x4058.x4050.rs.dns-oarc.net.
"195.113.187.254 sent EDNS buffer size 4096"
"Tested at 2015-06-11 11:28:59 UTC"
"195.113.187.254 DNS reply size limit is at least 4064"
```

3 Instalace rekurzivního resolveru

Cílem této úlohy je zprovoznit rekurzivní resolver.

- Vyzkoušejte instalaci resolverů Unbound a BIND.
- Zprovozněte a vyzkoušejte DNSSEC validaci.
- Omezte rozsah adres, pro které bude server sloužit jako otevřený rekurzivní server.

3.1 Instalace serveru Unbound

```
# apt-get install unbound unbound-host
…

# unbound-control status
version: 1.4.22
verbosity: 1
threads: 1
modules: 2 [ validator iterator ]
uptime: 49 seconds
unbound (pid 960) is running...
```

Výchozí konfigurace v Debianu automaticky stáhne *trust anchor* pomocí utility `unbound-anchor` a umístí do `/var/lib/unbound/root.key`. Jeho použití je automaticky nakonfigurováno v souboru `/etc/unbound/unbound.conf.d/root-auto-trust-anchor-file.conf`, takže hned od instalace je prováděna DNSSEC validace. Klíč je nadále udržován aktuální unboundem, je tedy nutné, aby soubor s klíčem byl unboundem zapisovatelný.

Pro další konfiguraci nahlédneme do dokumentace:

```
# less /usr/share/doc/unbound/examples/unbound.conf
```

V základní konfiguraci Unbound poslouchá jen na loopbacku. Změníme to a zavedeme omezení adres, které mohou pokládat dotazy, vytvořením souboru `/etc/unbound/unbound.conf.d/access-control.conf`:

```
server:
        interface: 0.0.0.0
        interface: ::

        access-control: 0.0.0.0/0 refuse
        access-control: ::0/0 refuse
        access-control: 127.0.0.0/8 allow
        access-control: ::1 allow
        access-control: 195.113.0.0/16 allow
        access-control: 2001:718::/32 allow
```

Pro změnu rozhraní, na kterých bude server poslouchat, je třeba jej restartovat příkazem `systemctl restart unbound`. Ostatní změny je možné provádět pomocí ovládací utility `unbound-control`. Tou je také možné vyčistit cache a upravovat některé části konfigurace za chodu. Vyzkoušejte nastavit přeposílání všech dotazů na server 8.8.8.8:

```
# unbound-control forward 8.8.8.8
ok
```

Zkontrolujte, zda správně funguje validace provedením dotazů na `www.dnssec.cz` (odpověď musí obsahovat flag `ad`) a `www.rhybar.cz` (musí vrátit návratový kód `SERVFAIL` v prvním případě a nevalidovaná data spolu s flagem `cd` v druhém případě).

```
# dig @localhost www.dnssec.cz
# dig @localhost www.rhybar.cz
# dig @localhost www.rhybar.cz +cdflag
```

Pomocí utillity `unbound-host` lze snadno získat detailní informace o příčině selhání validace. Ve výchozím stavu ovšem utilita nemá žádný pevný bod důvěry. Ten může načíst buď přímo, nebo z konfiguračního souboru serveru unbound:

```
# unbound-host www.rhybar.cz
www.rhybar.cz has address 217.31.205.55
www.rhybar.cz has IPv6 address 2001:1488:0:3::7
# unbound-host -f /var/lib/unbound/root.key www.rhybar.cz
# unbound-host -C /etc/unbound/unbound.conf www.rhybar.cz
www.rhybar.cz has address 217.31.205.55
validation failure <rhybar.cz. A IN>: no keys have a DS with algorithm RSASHA1
from 2001:678:11::1 for key rhybar.cz. while building chain of trust
```

3.2 Instalace serveru BIND

```
# apt-get purge unbound
# apt-get install bind9
```

V základním nastavení je rekurze povolena jen pro loopback a přímo připojené lokání sítě. Změníme to editací v sekci options konfiguračního souboru, která je v Debianských systémech oddělena do souboru `/etc/bind/named.conf.options`.

```
options {
…
allow-recursion { localhost; my_networks; };
};

acl my_networks {
2001:718:1::/48;
195.113.219.0/24;
};
```

Po úpravě konfigurace zkontrolujeme nastavení:

```
# named-checkconf
# systemctl restart bind9
```

DNSSEC validace je automaticky zapnuta volbou `dnssec-validation auto`. V takovém případě se pevný bod důvěry při prvním spuštění ověří podle klíče vestaveného při kompilaci a dále se udržuje v souboru `/var/cache/bind/managed-keys.bind`. Opět zkontrolujeme funkčnost validace:

```
# dig @localhost www.dnssec.cz
# dig @localhost www.rhybar.cz
# dig @localhost www.rhybar.cz +cdflag
```

4 Autoritativní DNS server

- Vytvořte zónový soubor pro zónu `zone.XX.acad.cz`
- Nakonfigurujte vybraný autoritativní server a ověřte jeho funkci.
- Vložte do souboru IP adresy svého serveru jako `ns.zone.XX.acad.cz`
- Vytvořte delegaci z nadřazené zóny a ověřte její funkčnost.

Pro přehlednost shrnuje názvy zón a jména serverů, na kterých jsou hostovány, níže uvedená tabulka. Zóna `XX.acad.cz` je na pomocném serveru již připravena jako prázdná, vaším úkolem je tedy vytvořit subdoménu `zone` na svém hlavním serveru a na serveru `nXXa` provést správně delegaci.

zóna	jméno serveru
`XX.acad.cz`	`nXXa.nebula.cesnet.cz`
`zone.XX.acad.cz`	`nXX.nebula.cesnet.cz` a/nebo `ns.zone.XX.acad.cz`

4.1 Příklad zónového souboru

Poznámka: Časovací parametry zde uvedených příkladů byly záměrně sníženy. V produkčním nasazení je nepoužívejte.

```
$ORIGIN zone.XX.acad.cz.
$TTL 60

@ IN SOA nXX.nebula.cesnet.cz. hostmaster (
                                1           ; serial
                                120         ; refresh (2 minutes)
                                10          ; retry (10 seconds)
                                3600        ; expire (1 hour)
                                60          ; minimum (1 minute)
                                )
@ IN NS nXX.nebula.cesnet.cz.
```

4.2 Příklady definice zóny

4.2.1 BIND

```
zone "zone.XX.acad.cz" {
  type master;
  file "/etc/bind/zone.XX.acad.cz";
};
```

4.2.2 Knot

```
zones {
  zone.XX.acad.cz { file "/etc/knot/zone.XX.acad.cz"; }
}
```

4.2.3 NSD

```
zone:
  name: zone.XX.acad.cz
  zonefile "/etc/nsd/zone.XX.acad.cz"
```

4.3 Přidání záznamu do zónového souboru

1. Vložte do zónového souboru záznam `ns.zone.XX.acad.cz.`
2. Zvyšte sériové číslo zóny.
3. Načtěte novou verzi zóny do DNS serveru (`rndc reload`, `knotc reload`, `nsd-control reload`)

4.4 Vytvoření delegace

4.4.1 Bez glue záznamů

Toto je jednodušší varianta. Zóna je nadelegována na server, který leží mimo zónu samotnou. V našem případě leží server zóny `zone.XX.acad.cz` v zóně `cesnet.cz`. Proto stačí vložit do nadřazené zóny stejný NS záznam, který je v apexu zóny.

```
$ ssh -lroot nXXa.nebula.cesnet.cz
root@nXXa:~# cat >> /etc/knot/XX.acad.cz
zone 60 IN NS nXX.nebula.cesnet.cz.
<Ctrl+D>
root@nXXa:~# knotc reload
OK
root@nXXa:~# logout
```

Poznámka: Pro jednoduchost zde při změně obsahu zóny needitujeme sériové číslo v SOA záznamu. Můžeme si to dovolit jednak proto, že server nemá žádné slave servery, jednak proto, že Knot DNS v režimu automatického DNSSEC podepisování sériové číslo zvyšuje automaticky.

4.4.2 S glue záznamy

Leží-li DNS server uvnitř zóny, která na něj má být nadelegována, samotný NS záznam v nadřazené zóně nestačí. Nadelegujeme tedy server zóny `zone.XX.acad.cz` na adresu `ns.zone.XX.acad.cz`. Změnu delegace na serveru `nXXa` tentokrát provedeme pomocí dynamického DNS.

```
$ ssh -lroot nXXa.nebula.cesnet.cz
root@nXXa:~# nsupdate
> server localhost
> update delete zone.XX.acad.cz.
> update add zone.XX.acad.cz. 60 IN NS ns.zone.XX.acad.cz.
> update add ns.zone.XX.acad.cz. 60 IN A 78.128.211.XX
> update add ns.zone.XX.acad.cz. 60 IN AAAA 2001:718:1:1f:50:56ff:feee:XX
> show
Outgoing update query:
;; ->>HEADER<<- opcode: UPDATE, status: NOERROR, id:      0
;; flags:; ZONE: 0, PREREQ: 0, UPDATE: 0, ADDITIONAL: 0
;; UPDATE SECTION:
zone.XX.acad.cz.        0       ANY     ANY
zone.XX.acad.cz.        60      IN      NS      ns.zone.XX.acad.cz.
ns.zone.XX.acad.cz.     60      IN      A       78.128.211.XX
ns.zone.XX.acad.cz.     60      IN      AAAA    2001:718:1:1f:50:56ff:feee:XX

> send
> quit
root@nXXa:~# logout
```

5 Master – slave replikace

- Nastavte svůj server tak, aby umožňoval zónové přenosy:
  1. každému
  2. vybraným IP adresám
  3. každému, kdo použije správný TSIG klíč
- Přidejte do konfigurace svého serveru slave zónu pro svého kolegu.
- Upravte NS záznamy v apexu zóny i v delegaci tak, aby vaše zóna byla dostupná i v případě výpadku vašeho serveru.
- Nastavte posílání zpráv `NOTIFY` pro rychlou synchronizaci slave serverů a zkontrulujte jejich správnou funkci editací zóny na master serveru.

5.1 Výroba a použití TSIG klíče

Je možné použít libovoný generátor náhodných čísel v base64 kódování o patřičné délce. Alternativou je nástroj `dnssec-keygen`:

Poznámka: Pro produkční účely **nepoužívejte** volbu `-r /dev/urandom`. Na strojích s nedostatkem entropie by takové klíče a podpisy nemusely být bezpečné.

```
# dnssec-keygen -a HMAC-SHA256 -b 256 -n HOST \
  -r /dev/urandom tsig.zone.XX.acad.cz.
Ktsig.zone.XX.acad.cz.+163+50384
# cat Ktsig.zone.XX.acad.cz.+163+50384.key
tsig.zone.XX.acad.cz. IN KEY 512 3 163 \
0su0ZsqZ6uxo3gMwGhzfKMNCBohnXYgURtJ2Dtf/zwM=
```

Ačkoli na identifikátoru klíče téměř nezáleží, je dobrou praxí pojmenovávat jej jako doménové jméno ve vaší správě. Takový klíč je následně nutné nakonfigurovat na obě zůčastněné strany: (příklad pro BIND)

```
key "tsig.zone.XX.acad.cz" {
  algorithm HMAC-SHA256;
  secret "0su0ZsqZ6uxo3gMwGhzfKMNCBohnXYgURtJ2Dtf/zwM=";
};

zone "zone.XX.acad.cz" {
  …
  allow-transfer { key "tsig.zone.XX.acad.cz"; };
};
```

Funkčnost TSIG přenosu je možné vyzkoušet i příkazem `dig`. Bez ohledu na správnost nebo nespránost klíče je tímto *trikem* možné vylákat ze vzdáleného serveru jeho systémový čas.

```
# dig zone.XX.acad.cz axfr -y HMAC-SHA256:tsig.zone.XX.acad.cz:\
0su0ZsqZ6uxo3gMwGhzfKMNCBohnXYgURtJ2Dtf/zwM= @nXX.nebula.cesnet.cz
…
tsig.zone.nxxa.acad.cz. 0       ANY     TSIG    hmac-sha256. 1434379000 …
```

6 DNSSEC na autoritativním serveru

- Vygenerujte ZSK a KSK klíče.
- Vložte klíče do zónového souboru.
- Podepište zónu pomocí nástroje `dnssec-signzone`.
- Vystavte otisk KSK klíče do nadřazené zóny.
- Zkontrolujte úplnost řetězu důvěry.
- Volitelně vyzkoušejte rotaci ZSK a KSK klíče.

6.1 Výroba klíčů

```
# apt-get install bind9utils
# cd /etc/{bind,knot,nsd}/
# ZSK=$(dnssec-keygen -a RSASHA512 -b 1024 -r /dev/urandom zone.XX.acad.cz)
# KSK=$(dnssec-keygen -a RSASHA512 -b 2048 -f KSK -r /dev/urandom \
  zone.XX.acad.cz )
```

6.2 Podpis zóny

```
# cat $ZSK.key $KSK.key >> zone.XX.acad.cz
# dnssec-signzone -r /dev/urandom -k $KSK -N unixtime -x zone.XX.acad.cz $ZSK
Verifying the zone using the following algorithms: RSASHA512.
Zone fully signed:
Algorithm: RSASHA512: KSKs: 1 active, 0 stand-by, 0 revoked
                      ZSKs: 1 active, 0 stand-by, 0 revoked
zone.XX.acad.cz.signed
```

Pro NSEC3 přidejte volbu `-3 <sůl>`. Nyní je třeba překonfigurovat DNS server, aby načetl zónový soubor s příponou `.signed`.

6.3 Vytvoření bezpečné delegace

Aby vznikl řetěz důvěry, je nutné do nadřazené zóny umístit DS záznamy. Ty jsou vygenerovány během podepisování do souboru `dsset-zone.XX.acad.cz.`, případně je možné je vygenerovat znovu:

```
# dnssec-dsfromkey $KSK
zone.XX.acad.cz. IN DS 55782 10 1 34F20A15D93B11D9448C57032EC4F6ABE8847B5B
zone.XX.acad.cz. IN DS 55782 10 2 98CD790243051453E588F4BB0939804DE7719096…66
```

Kterýkoli z těchto záznamů, případně oba, je potřeba vložit do nadřazené zóny:

```
$ ssh -lroot nXXa.nebula.cesnet.cz
root@nXXa:~# cat >> /etc/knot/XX.acad.cz
zone.XX.acad.cz. IN DS 55782 10 1 34F20A15D93B11D9448C57032EC4F6ABE8847B5B
<Ctrl+D>
root@nXXa:~# knotc reload
OK
root@nXXa:~# logout
```

6.4 Rotace ZSK metodou předpublikace

```
# ZSK2=$(dnssec-keygen -a RSASHA512 -b 1024 -r /dev/urandom zone.XX.acad.cz)
# cat $ZSK2.key >> zone.XX.acad.cz
# dnssec-signzone -r /dev/urandom -k $KSK -N unixtime -x zone.XX.acad.cz $ZSK
Verifying the zone using the following algorithms: RSASHA512.
Zone fully signed:
Algorithm: RSASHA512: KSKs: 1 active, 0 stand-by, 0 revoked
                      ZSKs: 1 active, 1 stand-by, 0 revoked
zone.nXX.acad.cz.signed
# # sleep $TTL + $REFRESH
# dnssec-signzone -r /dev/urandom -k $KSK -N unixtime -x zone.XX.acad.cz $ZSK2
# # sleep $TTL + $REFRESH
# # delete $ZSK from zone file and resign
```

6.5 Rotace KSK metodou dvojího podpisu

```
# KSK2=$(dnssec-keygen -a RSASHA512 -b 2048 -f KSK -r /dev/urandom \
  zone.XX.acad.cz )
# cat $KSK2.key >> zone.XX.acad.cz
# dnssec-signzone -r /dev/urandom -k $KSK -k $KSK2 -N unixtime -x \
  zone.XX.acad.cz $ZSK2
Verifying the zone using the following algorithms: RSASHA512.
Zone fully signed:
Algorithm: RSASHA512: KSKs: 2 active, 0 stand-by, 0 revoked
                      ZSKs: 1 active, 0 present, 0 revoked
# # sleep $TTL + $REFRESH
# # update parent DS record
# # sleep $parentTTL + $parentREFRESH
# # delete $KSK from zone file and resign
```