

Zone generation, DNS, DNSSEC

Jaromir Talir • jaromir.talir@nic.cz • 28.05.2014



Agenda

- DNS servers
 - Knot DNS
- Zone file generation
 - Cron
- DNSSEC basics
 - Key generation
 - Zone signing



DNS servers

- Authoritative DNS servers
 - Provide authoritative DNS data
 - Administrated by domain operators
- Recursive DNS servers
 - Provide cached DNS data
 - Administrated by ISPs



Authoritative DNS servers

- Bind 9.10 – ISC
- Bind 10 – Abandoned by ISC
- NSD 4.0.3 – NLNet labs
- Knot DNS 1.4.6 – CZ.NIC
- Yadifa - EURid



Knot DNS

- <http://www.knot-dns.cz>
- Packages for Ubuntu, Fedora
- Used in .CZ, .DK and some others
- Very high response rate
- Non-stop operation



Knot DNS – Ubuntu installation

- `add-apt-repository ppa:cz.nic-labs/knot-dns`
- `apt-get update`
- `apt-get install knot`



Knot DNS - /etc/knot/knot.conf

```
remotes {
    slave0 { address 0.0.0.0/0; }
}

system { rundir "/var/run/knot"; }

interfaces {
    any-ipv4 { address 0.0.0.0@53; }
    any-ipv6 { address [::]@53; }
}

control { listen-on "knot.sock"; }

log { syslog { any warning, error; } }

zones {
    storage "/var/lib/knot";
    cz {
        file "db.cz";
        xfr-out slave0;
    }
}
```



Knot DNS

- Start DNS server
 - knotc start
- Next steps
 - Configure transfers to your slave servers
 - Secure transfers with TSIG – transfers are signed with shared secret



Zone file generation

- Primary output of registry software is **the zone file**
- Many tools for checking validity of zone
 - named-checkconf (bind-utils)
 - Idns-verify-zone (Idns)
 - <http://www.nlnetlabs.nl/projects/credns/>
- Check number of changes from versions



Zone file generation - cron

- Cron – tool to automate repeating tasks
- Jobs are stored at **/etc/crontab** file:

```
# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12)
# | | | | .----- day of week (0 - 6) (Sunday=0 or 7)
# | | | | |
# * * * * * user-name  command to be executed
```



Zone file generation - cron

- Either edit **/etc/crontab** directly
- Or run **crontab -e**
- Add following row to do zonefile generation every minute

```
* * * * * root /usr/bin/genzone-client -z /var/lib/knot/
```

- By default, **gezone-client** makes copy of every zonefile with timestamp so be cautious about disk space



DNSSEC

- DNS is based on simple transactions
 - ServerA->ServerB: Do you know the answer?
 - ???->ServerA: This is the answer
- DNSSEC is protection against spoofing responses
- Every DNS query is signed and this brings:
 - Authenticity – the sender is not spoofed
 - Integrity – nobody inserted fake data during transport



DNSSEC

- Based on asymmetric cryptography
- One side signs with private key
 - Owner of domain on authoritative DNS server
- Other side verifies with public key
 - Regular user through properly configured recursive DNS server provided by ISP
- Public keys are verified through chain of trust



DNSSEC

- Managing DNSSEC keys
- Signing zone
- Sending public part of key to root zone
- Accepting public part of key from subsidiary zones

- Managing DNSSEC keys and signing can be automated



Management of DNSSEC keys

- The best practice is to have two keys
- ZSK – zone signing key
 - Weaker, rotated more often, used to sign every record
- KSK – key signing key
 - Stronger, rotated less often, used only to sign keys, published to upper zone



Management of DNSSEC keys

- Where to store keys
 - HSM – hardware security module
 - SoftHSM – software version
 - Filesystem
- Key rotation
 - Public part of keys may be subject of attack
 - Rotation involves complex process because of DNS caching and cooperation with upper zone



Management of DNSSEC keys

- Keys creation using dnssec-keygen (b
- KSK
 - `dnssec-keygen -f KSK -a RSASHA256 -b 4096 -r /dev/urandom cz`
- ZSK
 - `dnssec-keygen -a RSASHA256 -b 1024 -r /dev/urandom cz`



Signing zone file

- Signing using `dnssec-signzone`
 - `dnssec-signzone -r /dev/urandom -S db.cz`
- Creates new records in zone file
 - RRSIG with signatures
 - NSEC to fill spaces
- RRSEG signatures has validity
 - Resigning is necessary otherwise zone is **invalid!!**



Sending public part to upper zone

- As part of zone signing process DS records are created
- Send IANA request to include DS records in root zone



DNSSEC

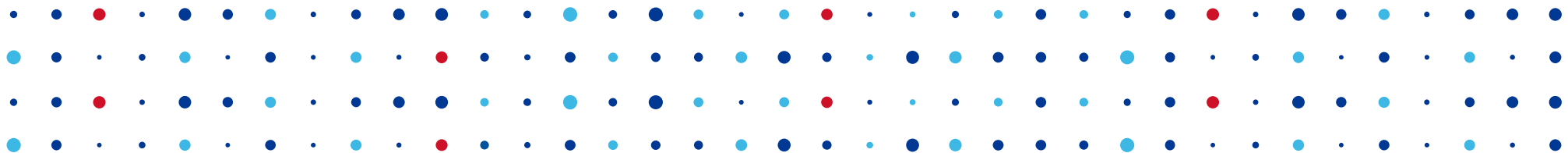
- Google for more information :)
- Check automated tools
 - OpenDNSSEC
 - BIND has automated signing
 - Knot DNS has automated signing



Accepting public keys

- You should allow registrants to include their public keys under your TLD
- It's responsibility of registry software to accept these data via EPP
- Two possibilities
 - DS records – hash of key and domain name
 - DNSKEY records – key itself





Thank You

Jaromir Talir • jaromir.talir@nic.cz

