DENIC DNSSEC Testbed
Software support for DNSSEC

Ralf Weber
(ralf.weber@nominum.com)

Who is Nominum?

Mission

- ***Deliver the Trusted Internet Experience***
- ***Strategic Partners:***

YAHOO!

EQUINIX

Sun microsystems

Qwest

Global Crossing

NTT Communications

Product Leadership

- ***Best DNS Security***
- ***Highest Scalability***
- ***Highest Reliability***
- ***All Open Standards***
- ***Pioneered Intelligent DNS***

Enabling rules and policies for every DNS request to protect end-users and ensure they reach their intended destination

Industry Expertise

- ***Dr. Paul Mockapetris***

Inventor of DNS, IETF Chair: 1994-1996
Lifetime award: ACM SIGCOMM 2005

- ***Bob Halley***

Co-Architect of BIND8
Architect of BIND9

- ***Ted Lemon***

Developer of ISC-DHCP
Co-author of DHCP Handbook

- ***Over 30 Standards authored or co-authored***

Securing the Worlds' Largest Carriers DNS Infrastructure with Over 170M Broadband Households

BT

verizon

orange

NTT Communications

TELECOM ITALIA

COLT

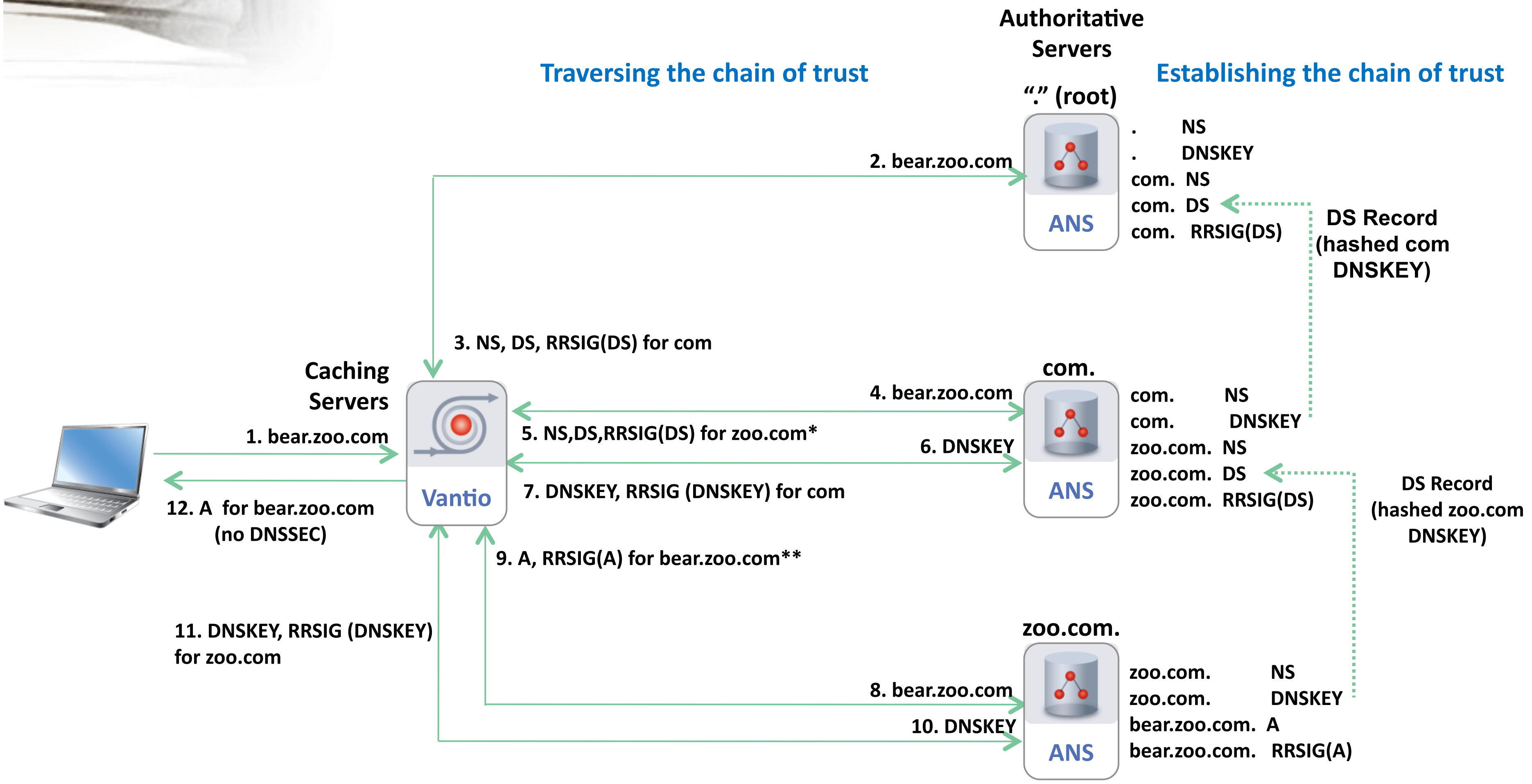
comcast

Deutsche Telekom

Telefónica

Virgin media

DNSSEC in one slide



Traversing the chain of trust

Establishing the chain of trust

Authoritative Servers

Caching Servers

1. bear.zoo.com

2. bear.zoo.com

3. NS, DS, RRSIG(DS) for com

4. bear.zoo.com

5. NS,DS,RRSIG(DS) for zoo.com*

6. DNSKEY

7. DNSKEY, RRSIG (DNSKEY) for com

8. bear.zoo.com

9. A, RRSIG(A) for bear.zoo.com**

10. DNSKEY

11. DNSKEY, RRSIG (DNSKEY)
for zoo.com

12. A for bear.zoo.com
(no DNSSEC)

"." (root)

. NS
. DNSKEY
com. NS
com. DS
com. RRSIG(DS)

DS Record
(hashed com
DNSKEY)

com.

com. NS
com. DNSKEY
zoo.com. NS
zoo.com. DS
zoo.com. RRSIG(DS)

DS Record
(hashed zoo.com
DNSKEY)

zoo.com.

zoo.com. NS
zoo.com. DNSKEY
bear.zoo.com. A
bear.zoo.com. RRSIG(A)

If verification is successful the DNS cache is populated with the A record, otherwise SERVFAIL is returned to clients

* An appropriate NSEC record and RRSIG(NSEC) are sent if the domain does not exist or is not signed

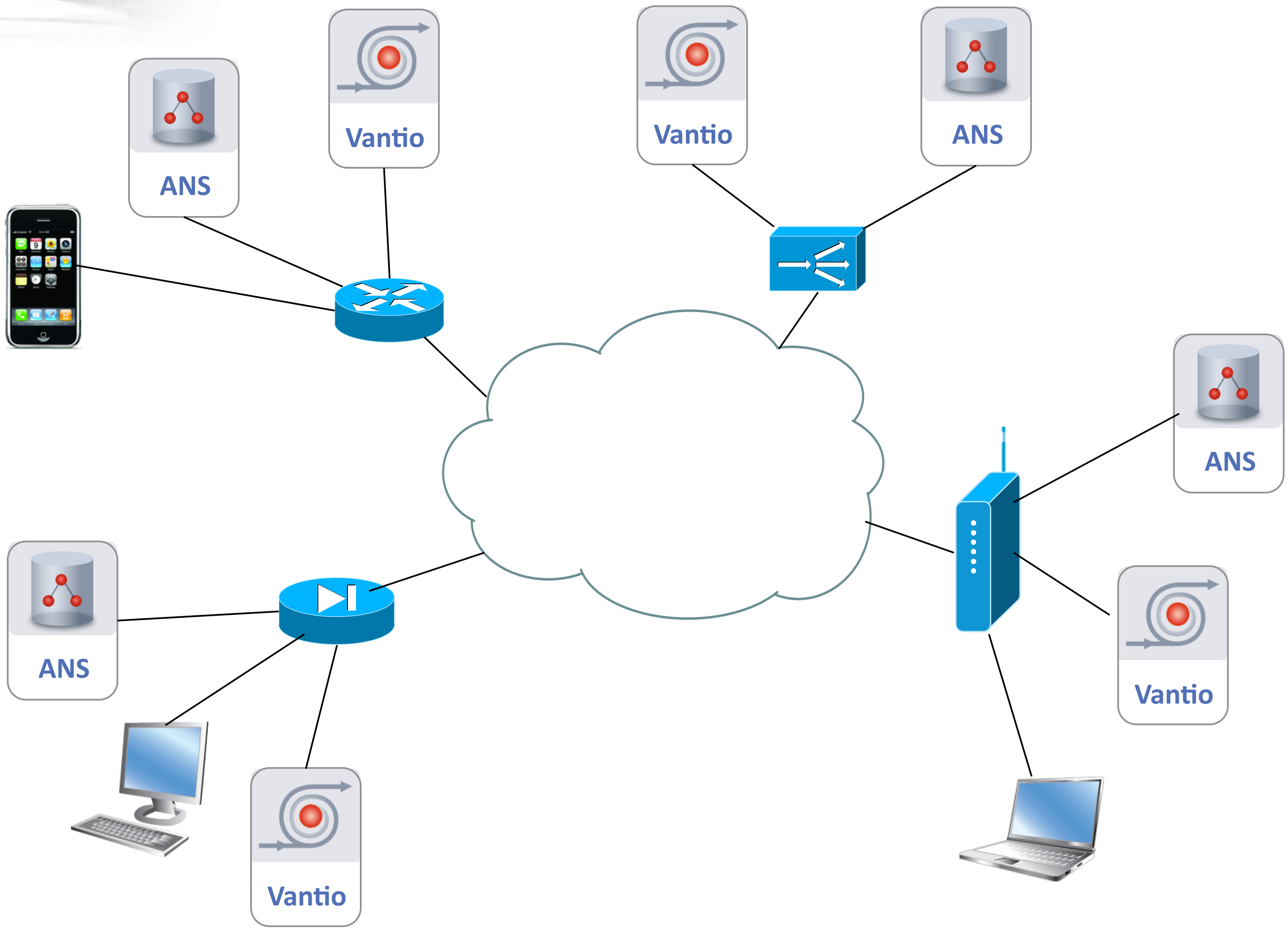
** An appropriate NSEC record and RRSIG(NSEC) are sent if the domain does not exist

What can go wrong

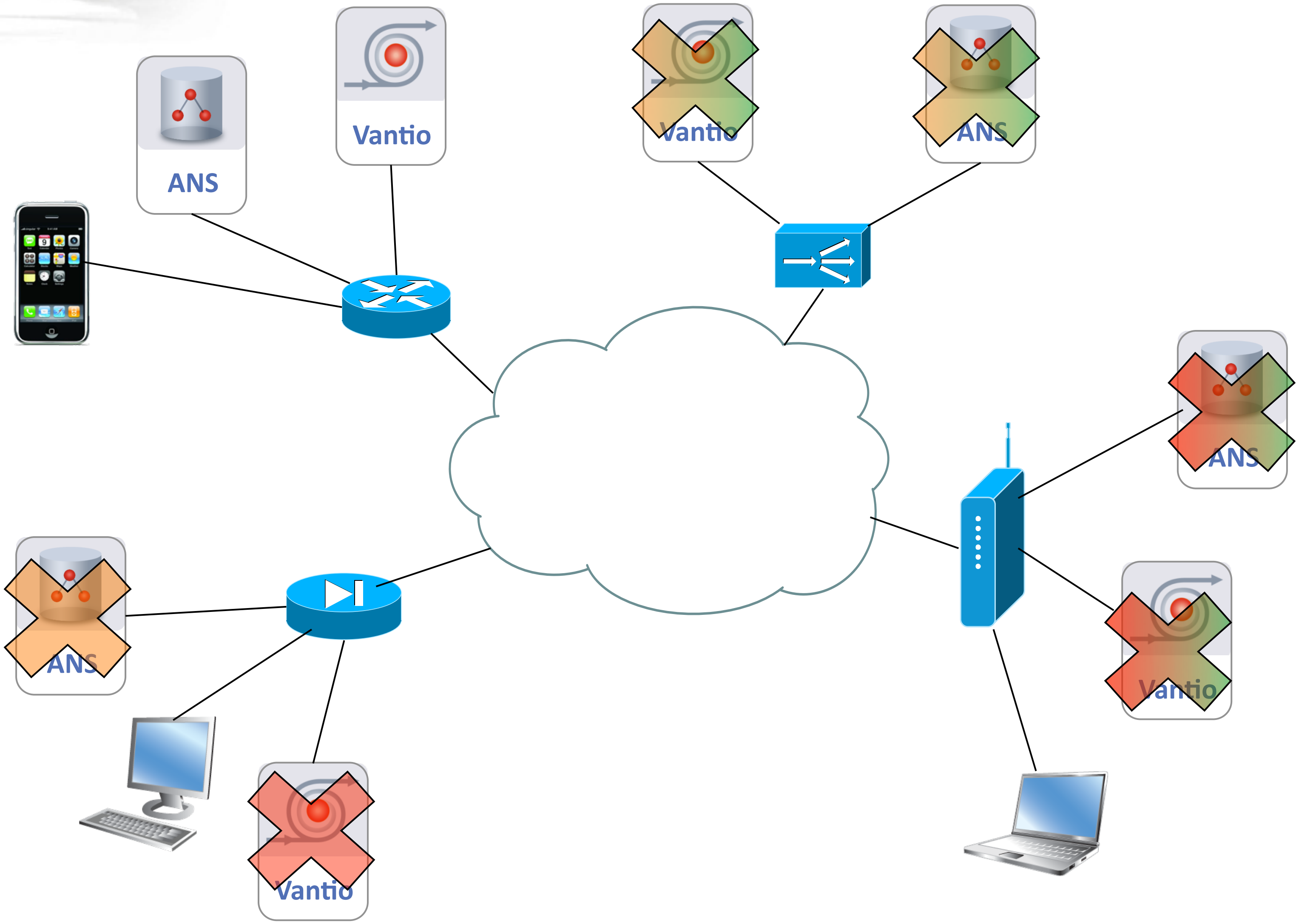
Nominum.

- Every error in the chain of trust cause resolutions to fail
- Cryptography requires constant changes
  - Signatures and keys have limited lifetimes
  - DNS data becomes dynamic with static content
  - Cryptographic algorithm may change
- Software has to be kept up to date or may fail
- DNS Data becomes bigger
  - A lot of people still believe DNS packets have a maximum size of 512 Bytes and UDP only
  - DNS UDP packets with EDNS0 can get bigger and fragment
  - If that’s not enough DNS will switch to TCP
  - Not all network devices might understand this

DNS and network devices



DNSSEC network problems

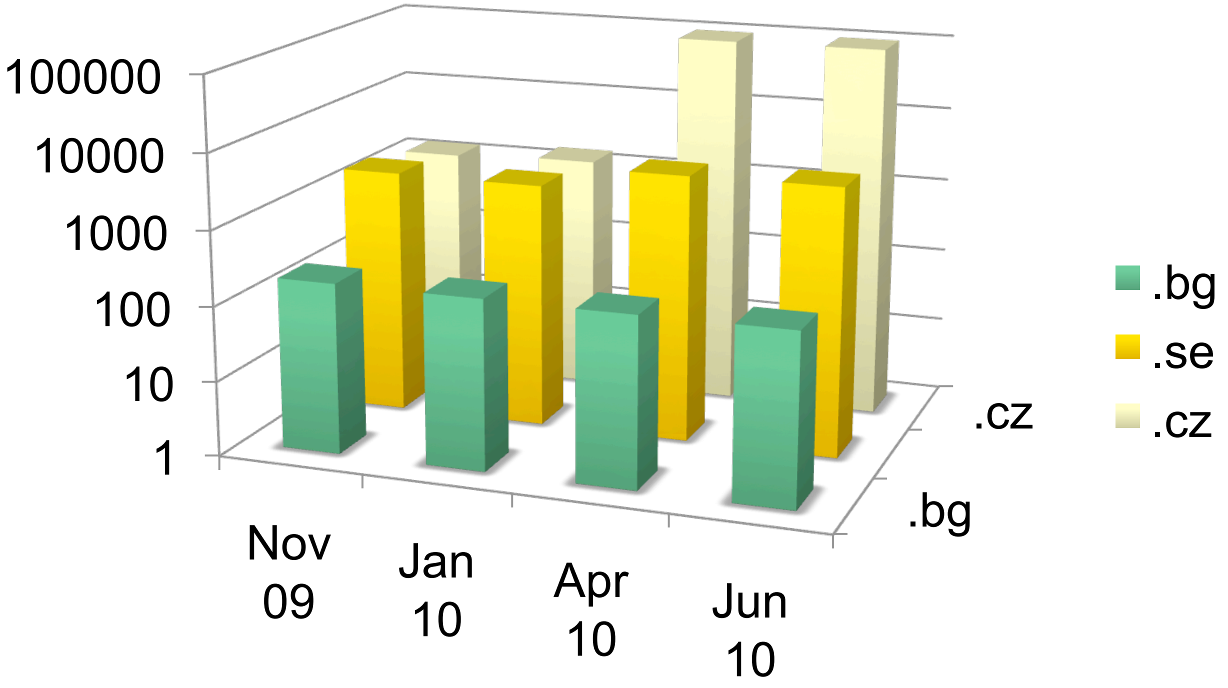


DNSSEC and the network

- Clients are fine
  - They don’t do DNSSEC validation at the moment
  - Windows and MacOSX don’t have a validator
  - Only Fedora has and they screw it
  - The home gateway (9 out of 38) discussion only affects geeks
  - Home gateways have gotten better (Thanks AVM)
- Don’t run DNS servers behind firewalls
  - It is possible but it usually requires configuration
  - Firewalls are not made for high qps throughput (to much state)
  - They often break DNS servers defenses
- Load balancers should not alter DNS packets
  - Mostly applies for Global Server Load Balancing
  - You can use them for pure load distribution

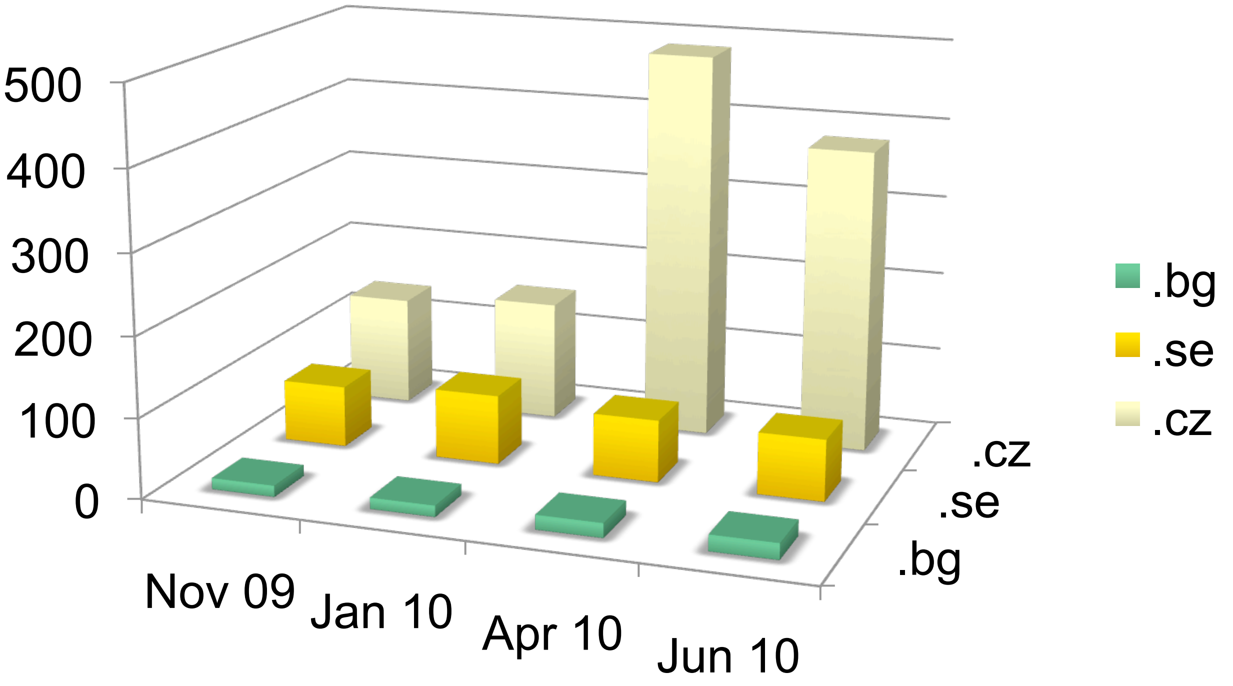
Some DNSSEC statistics

- Number of DNSSEC domains (log scale)



Some DNSSEC statistics

- Number of Domains that fail validation



Statistics Summary

- DNSSEC is gaining momentum
  - It’s good to see some large registrar taking it in CZ.
  - Some problems they might think about
    - All signatures expire at same time
    - Do not resign or roll everything at once
- Validation failures will be a problem
  - We need to get operators the tools to mitigate them
  - An insecure domain that resolves might be better than no resolving
  - Who would customers call when amazon.com failed

Validation failures

- How do validation failures happen ?
  - The data on the authoritative side is wrong
    - Signatures expired (arpa)
    - New keys without DS delegation at parent
    - Domain owner doesn't care about DNSSEC any longer (register.bg ;-)
- What can we do that they not happen ?
  - Don't require 70 pages documents for people to setup DNSSEC
  - Make the operator interface the same as it used to be
  - Automate the resigning
  - Automate the key rollover
  - Automate the parent/child key relationship

Nominum products for DNSSEC

Nominum.

- All our software has been supporting DNSSEC for years
- We support NSEC, NSEC3 and all production algorithms
- Different software for caching and authoritative functions
- Vantio for DNS caching services
  - Fastest caching server with or without DNSSEC
- ANS for DNS authoritative services
  - In memory versioning database
- Configuration
  - All configuration is done on the running server and instantly active
  - No restart or file reload necessary

DNS Caching Server Challenges with DNSSEC

- Seamless resolution of signed and unsigned zones
  - Validation enabled for all domains under a defined trust anchor
  - Add one line to configuration for ITAR
    - trust-anchor-file "/var/nom/vantio/anchors.mf";
  - Possible to add more keys for islands of trust
    - trusted-keys { a0.com.invalid. 257 3 5 \ "AQO6CI+sIAf +iuieDim9L3kujFHQD7s/IOj03ClMOpKYcTXtK4mRpuUL VfvWxDi9Ew/ gj0xLnnX7z9OJHIxLI+DSrAHd8Dm0XfBEAtVtJSn70GaPZ gnLMw1rk5ap2DsEoWk=" };
  - Possible to remove domains from validation if domain owners screw it
    - negative-trust-anchors { arpa.; register.bg.; };

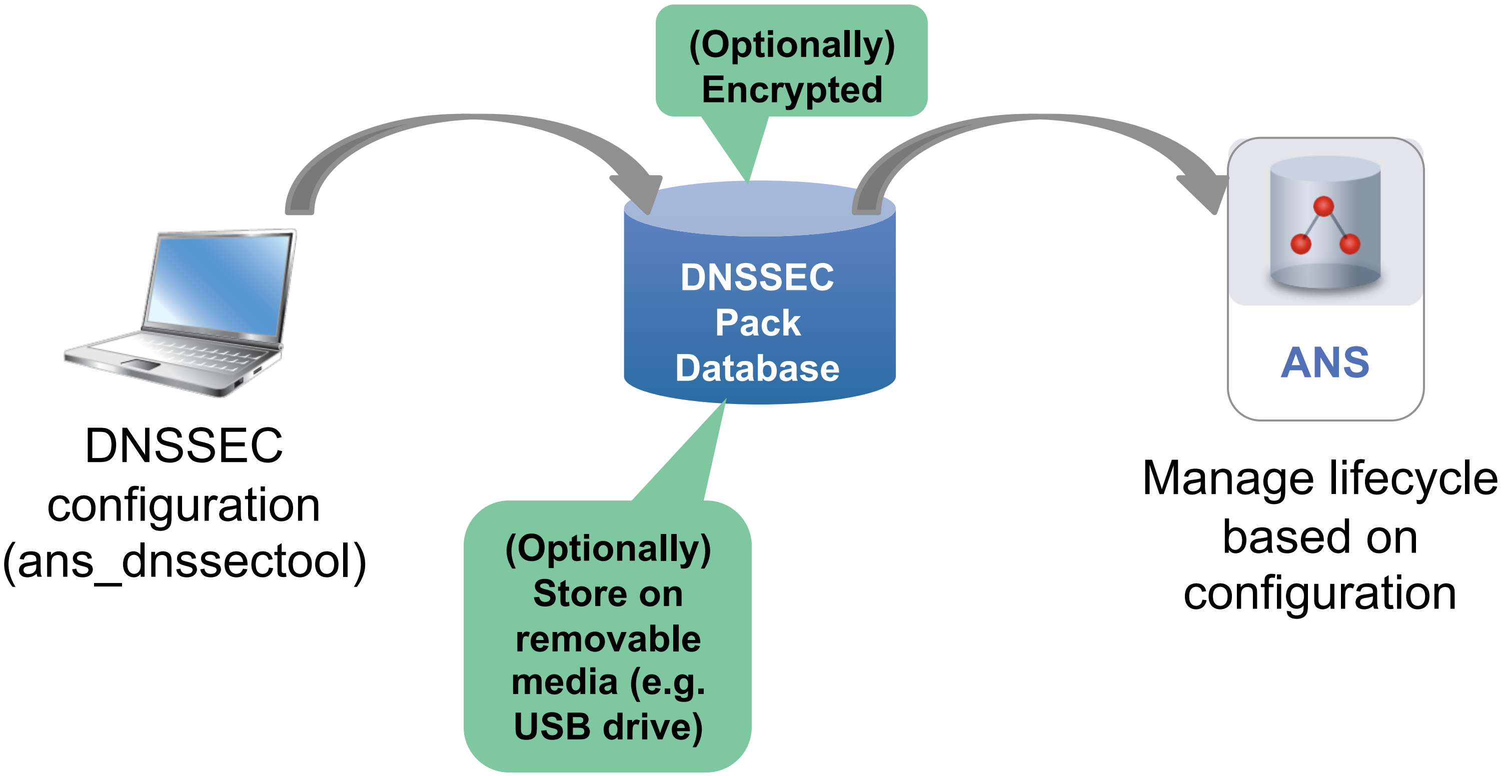
Authoritative Server Challenges with DNSSEC

- Signing/resigning zones is CPU intensive
  - ANS leverages multi-core CPUs to sign most zones online, but out of ‘fast path’
- Database size can increase by 6x or more
  - ANS uses optimized database technology to handle large increases in data required by DNSSEC
- Key Administration
- Managing Signing of Zones
- Updating Zones When Data Changes
  - Manual zone file re-signing when records are added, changed or deleted from a zone (via DDNS or edits)

Solution is Nominum DNSSEC Packs

Nominum DNSSEC Packs

- An administrative bundle that manages DNSSEC lifecycle automation
  - Automatically sign/resign zones online
  - Automatically rollover keys (e.g. update after 60 days) based on policy
- All activities done via single command utility (“ans_dnssectool”)



(Optionally) Encrypted
DNSSEC Pack Database
ANS
DNSSEC configuration (ans_dnssectool)
(Optionally) Store on removable media (e.g. USB drive)
Manage lifecycle based on configuration

Securing a zone

- To secure a previously insecure zone, create a pack for it.
  - ans_dnssectool create-pack --name initial example.com

- What this does:
  - Creates a KSK for the zone. The default is a 2048-bit RSA/SHA1 key.
  - Creates a ZSK for the zone. The default is a 1024-bit RSA/SHA1 key.
  - Gives the initial signing-data the name "initial“ (used in logging)

- Result:
  - The server will immediately begin signing the zone
  - Publishes it when signing completes
  - Server logs the publishing progress
  - Automatically resigns zone before signatures expire
  - New records are automatically signed with current keys

- Online signing support is key to allowing tools to handle signing as transparently as possible

What we mean by transparency
The hostmasters view

Insecure zone

```
@ 300 IN SOA  ( ns1 hostmaster
            1265702400
            3600
            600
            2592000
            300 )
@ 300 IN NS ns1
@ 300 IN NS ns2
ns1 300 IN A 192.0.2.1
ns2 300 IN A 192.0.2.2
www 300 IN A 192.0.2.3
```

Secure zone

```
@ 300 IN SOA  ( ns1 hostmaster
            1265702400
            3600
            600
            2592000
            300 )
@ 300 IN NS ns1
@ 300 IN NS ns2
ns1 300 IN A 192.0.2.1
ns2 300 IN A 192.0.2.2
www 300 IN A 192.0.2.3
```

And that is what other software or the wire give you

```
example.com.            300     IN      SOA     ns1.example.com.
      hostmaster.example.com. 1265702401 3600 600 2592000 300
example.com.            300     IN      RRSIG   SOA 5 2 300
      20100427203428 20100420172928 2790 example.com. RMzVVs/
      uV227uAbY9bMsVBTpEEAU5AI8OA0lSQ82/S1E96AK15JKQPOF
      OaUuIUwGLPf3UMO63sK2cx5SjkbRl7tQyVRD6T2dpVoSlBi75+ys1eKV
      HqE5e0cVVSYS7SZWdlLcpLEZ/fjBYlwqakFIBdaIWiCis1Ebmls7VZy9
      r7M=
example.com.            300     IN      NS      ns1.example.com.
example.com.            300     IN      NS      ns2.example.com.
example.com.            300     IN      RRSIG   NS 5 2 300
      20100427203428 20100420172928 2790 example.com.
      Zxt7LBFIExK2a+HV7e+E+noft1JRQfnB0ZOydM1v84Q9sNOR9/ioZQ+3
      21hOirE92fYrPj6Qe5fHWH+3Ti1PwWz65+JnvokulBHk3OPn+au7/CUc
      Va20jLAZ47vs7GmDLURnBN1OU/pes1pSbqoqDAtFjwoUrmcGtCwUAqe8
      YkI=
example.com.            300     IN      NSEC    ns1.example.com.
      NS SOA RRSIG NSEC DNSKEY
example.com.            300     IN      RRSIG   NSEC 5 2 300
      20100427203428 20100420172928 2790 example.com.
      SVAmmyja6s1du6nn8eQkYbfinjiVFpJXeWsmkarq0qqVHbfU9mkhmAqJ
      tGehQXNxduhkCBbyntd4XlIOxXm6lUEvEB7SbseJIgwAUh0Pni95Q8rx YFM
      +hJ+Bh7dTxubzoo1f+Jyhtk3jGUHR1Dn9y+d3i4122pzYoHfvPlhP KKA=
example.com.            3600    IN      DNSKEY  257 3 5
      AwEAAaEIqFpfKtDclyTsxFkudKjAnKq6bBfAbEG8SrlrhN8tryRRqOdE
      cdpMSrEfmGpjJWbKZ9i39tjbYcZnwCHyM/GpR96VCZtSuZAePoHOvU+x
      9hG5qCG/Luy45shp3UFkVvURCqevYj6uj7ru5uHsAYZewwzcQoUvmVgl
      aiKxFE+j8tH0PJF/+5BNArBxWS1gKRxrjLVcuSwoPteHzZ6ZLCGsqao2
      ak5FK9B3QX1hIOQ64TgAbkDlGbWf8pyY3NoXk5vcJlnXyvABrfAbnfog
      V7xm44JGaET8LniMJhrLEFlVW6Z0a0ytHUOAiN2cYw0P/mLGqqu9OAGJ
      Cxuu3y07bmU=
example.com.            3600    IN      DNSKEY  256 3 5
      AwEAAdeD9EWc5olFuUhbW0xp06Zb3C+Lym+8UrpjAB0kdtSTeXr7v5Ww
      fFQFUu8bU6aC6lJFnAa2sPyZTHSjk+t71nQAAbn3ILsQxjVMQEIYemRX
      rBYMK+/qkoDJUs/excAbePoLnry6joEZ4muSamu8nAl2nxFhm8jQC9Vn
      3LugB0ez
example.com.            3600    IN      RRSIG   DNSKEY 5 2 3600
      20100427203428 20100420172928 2790 example.com.
      HnJGACrWQDEiphiZPtJ5q2Ar01glwe8znrkq9uhnM5wr+NDGzQz93utt
      1MGrd6P9b81VgeIbCGMoc7E1dKfDc9uch4/mzMkDhDDszSDVS5zke84n
      9ZCKnRiz/4pNLkLW32ktNgsMT5/oJ2UXla2gspTgohu/CQi4ZZdnXv2k
      6ZY=
example.com.            3600    IN      RRSIG   DNSKEY 5 2 3600
      20100818173428 20100420133428 13426 example.com. N
      +UsDZ8B04S51Y6Ujt/o+MQ5HtxdkRQEaCNEpoMq6WG0QEUvxmrCWAvH
      cG9x9P12D0gJz36AS53cnrcdgMn5BePt6D/EXIhprO9eBtK+zpHaoNcQ
      a3bjIkz3J3heGiVirZ2y5OeXCXLY4J0w86c8dRpgm5J0W0YXVe0rAExp
```

```
      RRSIG NSEC
ns1.example.com.        300     IN      RRSIG   NSEC 5 3 300
      20100427203428 20100420172928 2790 example.com.
      Q6VyE0WGs7jUN5qder4f9WpVG9oWsaJ2v07FPwmIxa9uwcefISX6QgMN
      HIBsRA2YPLYBobNeN9TFMmAVpHPerG5UD45DA4hO2JwLptiU56D2o5AN
      FsQoTt4WEQ7o1L70NsZ+NfdXj+C6oKTJYlzIQ7u2dH1e2f1Y/yDwwZyl
      C44=
ns2.example.com.        300     IN      A       192.0.2.2
ns2.example.com.        300     IN      RRSIG   A 5 3 300
      20100427203428 20100420172928 2790 example.com. BbEKmp2Lb/
      Mt9cZtkQ/4H5rZQpy9sTPrEYcfjSKqf324gSd5abwWK47+
      VY1WT2WWo2WWXCW1Ir6gJgR5MUuIrw1gEaW7iMHhHctIaAdkDT0Z3gJT
      Fbl7TqfpiaA2g+xl5d9GdgN3B7EnpLpHZ2asTAmbRoO7F40JrTt+pZ7o
      baI=
ns2.example.com.        300     IN      NSEC    www.example.com. A
      RRSIG NSEC
ns2.example.com.        300     IN      RRSIG   NSEC 5 3 300
      20100427203428 20100420172928 2790 example.com.
      MXC2zhyPkQAWPFaL9Y/bZ5U9wDC0goHLa6MEU5nYsEZTjBe52Txxo1j/
      kxBCuv0TUfeTvbLc194rtJOO7MWlxK1v1mI0B13Vr8v2D9lTrYAT4px1
      IlaV2clQ2NVmI0ERFZSWeEEti4iBfXg2bBuAq2s/vzlEZ5SMqJSSCDV4
      GXo=
www.example.com.        300     IN      A       192.0.2.3
www.example.com.        300     IN      RRSIG   A 5 3 300
      20100427203428 20100420172928 2790 example.com.
      vAKUvf6lrNCyzuvwdyFD0j5YEpvm+KX9/85BlvyeGVmimRvgCciZRXt5
      fBgKgS1+4tqZ7iF2GaHsxsyfuFr4e3+z++efNSvgJPujh4bGKJXXg1lo
      RQWL2HNlocKeyY7hGhSxPX1hP+so7GRd4fZ2UDazQ5wiC7sSTX7xrL9l
      soQ=
www.example.com.        300     IN      NSEC    example.com. A
      RRSIG NSEC
www.example.com.        300     IN      RRSIG   NSEC 5 3 300
      20100427203428 20100420172928 2790 example.com.
      JnYMUFvVMKxoU9XWI+wD13oSzLkeh7b5QB88n4SKSF4QGZRseTOmCjzq /
      ntiWM1vIs4E3zs09y5eVrhB3E8O0GgUxdcMI2PaUSN0J1pdfHkl++yt
      bZhqTjIis+2cgd0qtjQX4JuvkiU1IOMLBcijEri28JP6rR5McurfWwNU
      0x4=
```

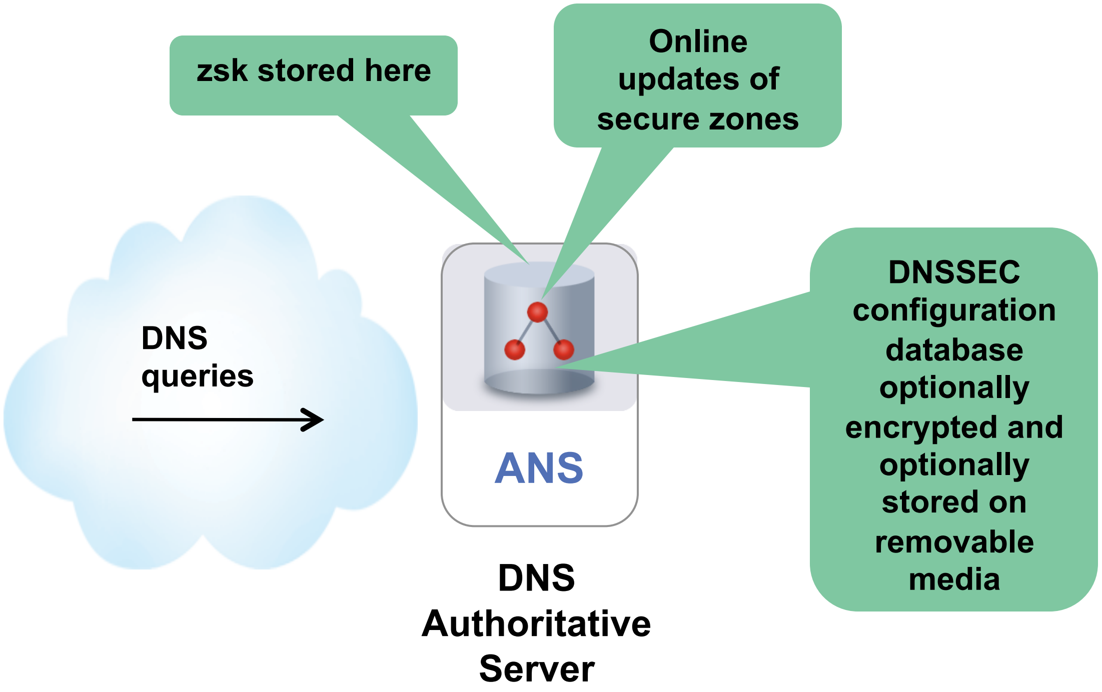
Example: ZSK rollover

- Periodically you will want to update your ZSK
  - All that is required as input is the time to do the update
- ans_dnssectool rollover-zsk --name autumn-zsk --start 20100715211800 example.com
- Signatures using the new ZSK will be published at 21:18 UTC on July 15, 2010.
  - New keys will be generated and start being used automatically
- You can provision multiple ZSK key rollovers that all will be stored in the database and executed appropriately

3 Levels of Security

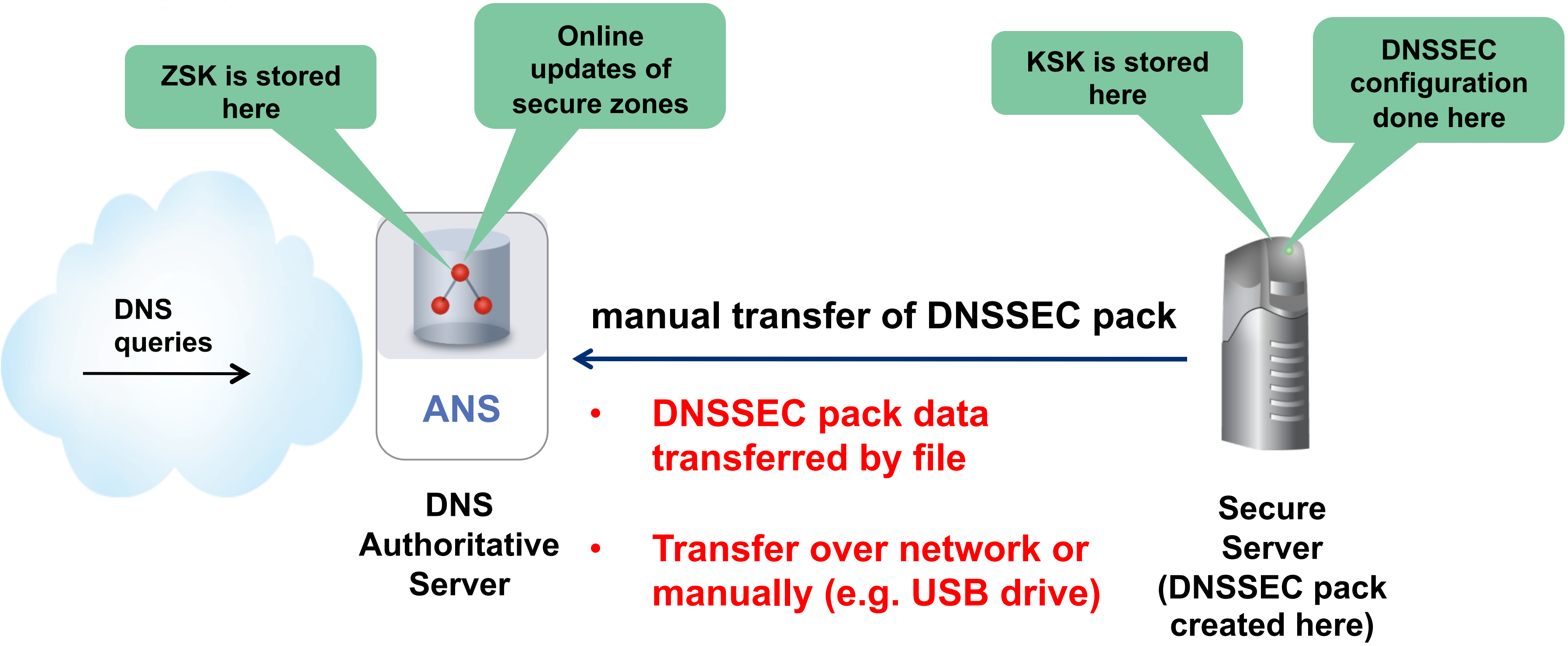
- Default Security
- Increased Security
- High Security

Default Security


zsk stored here
Online updates of secure zones
DNS queries
ANS
DNS Authoritative Server
DNSSEC configuration database optionally encrypted and optionally stored on removable media

- Private KSK not in readable format on disk
  - Optionally stored on removable disk (e.g USB drive)

Increased Security



ZSK is stored here

Online updates of secure zones

KSK is stored here

DNSSEC configuration done here

DNS queries

ANS

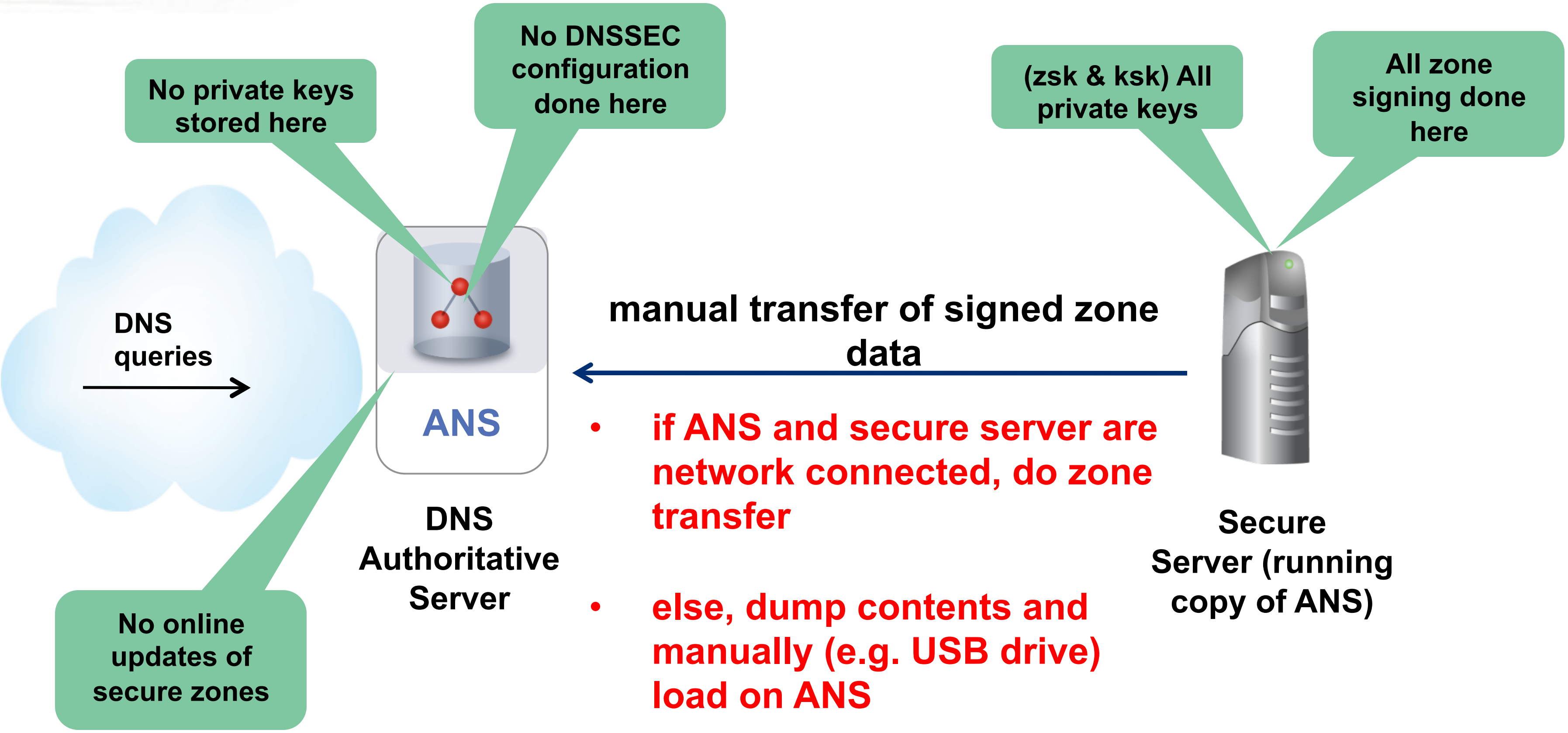
DNS Authoritative Server

manual transfer of DNSSEC pack

Secure Server (DNSSEC pack created here)

- DNSSEC pack data transferred by file
- Transfer over network or manually (e.g. USB drive)

High Security



No private keys stored here

No DNSSEC configuration done here

(zsk & ksk) All private keys

All zone signing done here

DNS queries

ANS

DNS Authoritative Server

manual transfer of signed zone data

- if ANS and secure server are network connected, do zone transfer
- else, dump contents and manually (e.g. USB drive) load on ANS

Secure Server (running copy of ANS)

No online updates of secure zones

A word on NSEC3

- NSEC3 is not better than NSEC
- It solves two problems most people don’t have
  - Data privacy for zones
  - Large delegation centric zones with only few secure delegations
- Data privacy is given by obfuscating the pointer to the next record
  - The next entry is not the name but a hash of the name
  - To make it even worse the hash can be called more than once
  - Computation of hash functions use CPU time
- Opt out NSEC3 records make validation a bit more complicated
  - They tell what parts of the zone are not secured
  - Validator has to check this

Questions?