DNSSEC

made easy

Inline signing with BIND

Secure Linux Administration Conference

Berlin
June 15 2016

Holger.Zuleger@hznet.de

Intro

- DNSSEC hat eine lange Geschichte (Und das ist nur der „neuere“ Teil)
  - 2001 Notes from the State-Of-The-Technology: DNSSEC (RFC3130)
  - 2005 DNSSEC Intro, Resource Records & Protocol Modifications (RFC4033, RFC4034, RFC4035)
  - 2006 Operational Practices (RFC4641)
  - 2007 Automated Updates of DNSSEC Trust Anchor (RFC5011)
  - 2008 DNSSEC Hashed Authenticated Denial of Existence (RFC5155)
  - 2012 Operational Practices, Version 2 (RFC6781)
  - 2014 Automating DNSSEC Delegation Trust Maintenance (RFC7344)
  - 2015 Key Rollover Timing (RFC7583)
- Entwicklung des Protokolls ist abgeschlossen
  „Nur“ noch Operational Hints and Modifications
- BIND Entwicklung begleitet die Protokoll Spezifikation
  - „Meine“ erste BIND Version mit DNSSEC: 9.3.0s20020618 (2002)
  - Neuste Version unterstützt (ansatzweise) CDS (RFC7344)

Operative Herausforderungen

- Jeder Resource Record Set benötigt eine Signatur (RRSIG)
  - RRSIG Record mit Start- & Endezeitpunkt (Gültigkeitsdauer)

```
www.ex.de.   IN A      1.2.3.4
             IN RRSIG A .. .. .. 20160617041931 20160607041931 ...
             IN AAAA   2001:db8:15::37
             IN RRSIG AAAA .. .. .. 20160617041931 20160607041931 ...
```

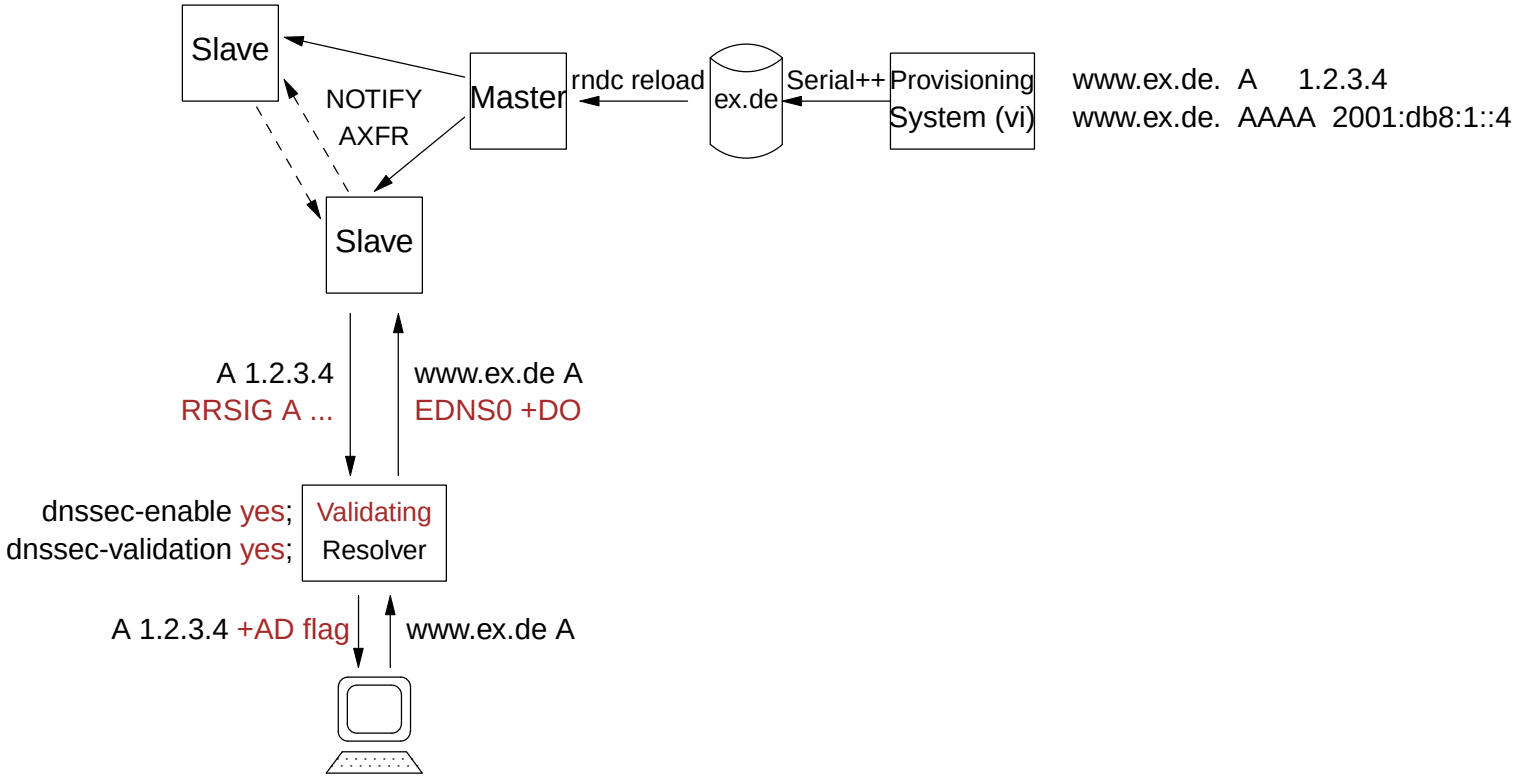
  - Regelmäßige Erneuerung essentiell
- Um Signaturen zu erzeugen benötigt man Schlüssel
  - Policy (Algorithmus, Keysize, Nutzungsdauer, etc.),
  - Key Signing Key (KSK) signiert ausschließlich DNSKEY RR
  - Zone Signing Key (ZSK) signiert alle (anderen) Resource Records
  - Key Rollover
- Schlüssel (Public Part) müssen veröffentlicht werden
  DNSKEY Record muß in der Zone stehen

```
ex.de.      DNSKEY      257 3 5 ... ; [KSK key id = 01837]
            DNSKEY      256 3 5 ... ; [ZSK key id = 64452]
```

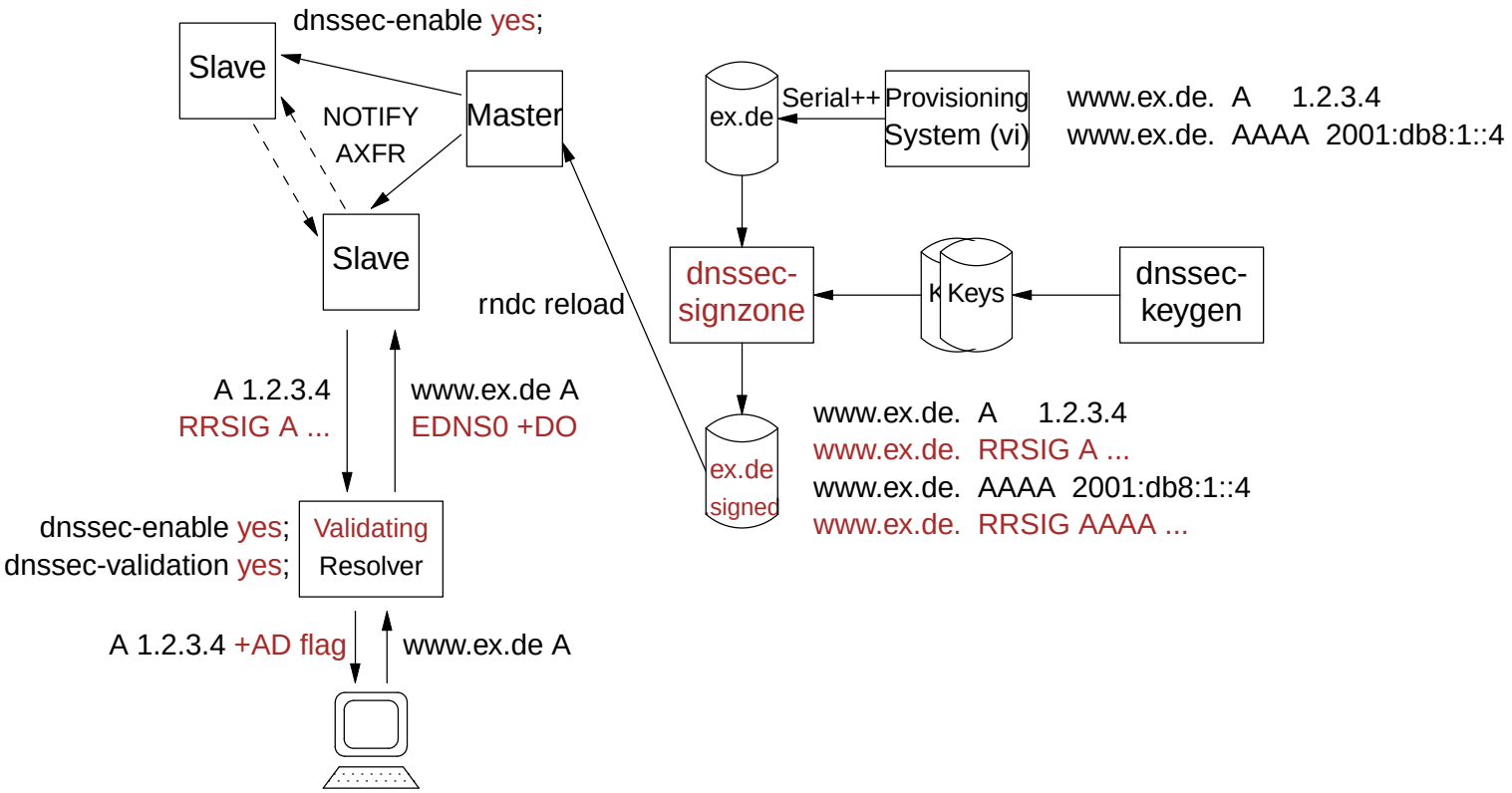
- Vertrauenskette aufbauen (Delegation Signer)
  DS Record in der Parentzone hinterlegen

```
ex.de.      DS          1837 5 1 ...
```

BIND Offline Signing



BIND Offline Signing



- Resigning notwendig auch wenn keine Änderung an der Zone
  Entkopplung der Seriennummer unsigned/signed Zone
- Inkrementelles Signieren möglich (`.signed` als Eingabedatei)
  Signaturen werden nicht zeitgleich erneuert (jitter)

BIND Konfiguration

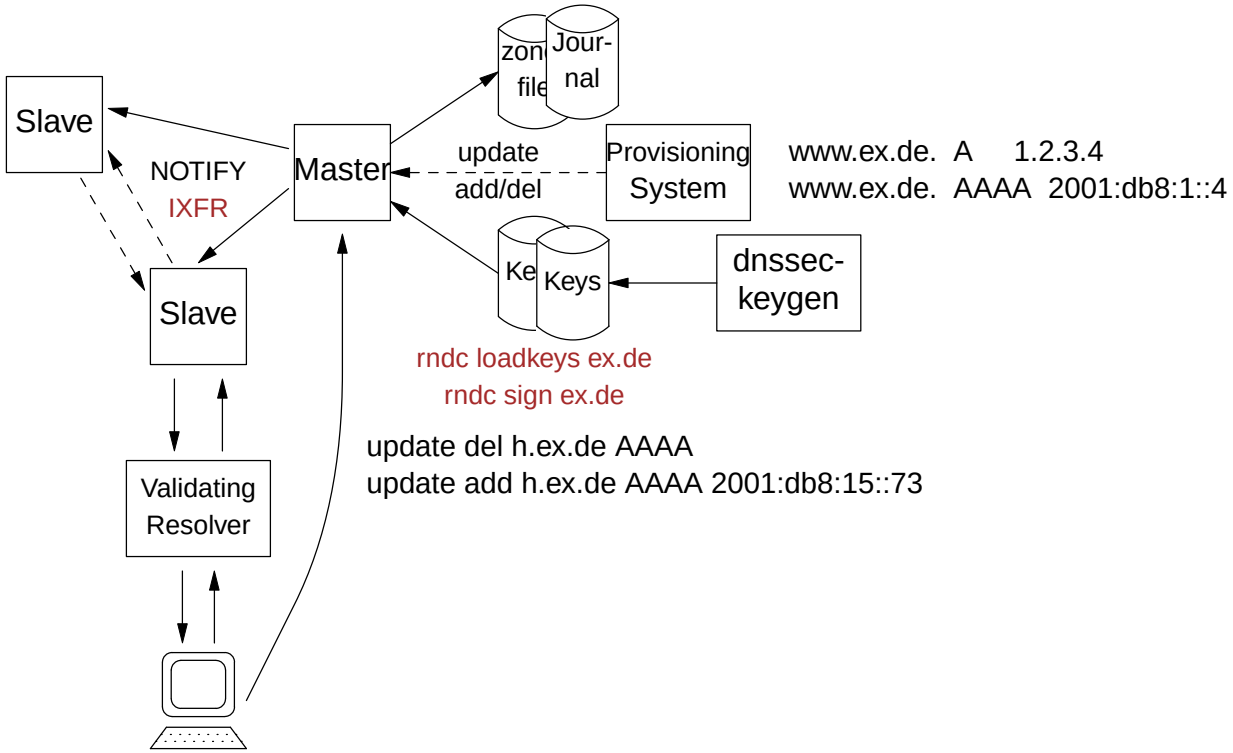
- Resolver

```
options {
        recursion yes;
        dnssec-enable yes;
        dnssec-validation yes;
        allow-recursion { recursive-acl; }; allow-query-cache { recursive-acl; };
};
managed-keys {
        "."  initial-key  257 3 8 "AwE....k1ihz0=" ; # key id = 19036
};
acl "rfc1918-acl"  {  10/8; 192.168/16; 172.16/12; fd00::/8;  };
acl "local-net"    {  2001:db8:356::/48; };
acl "recursive-acl" { 127.0.0.0/8; ::1; rfc1918-acl; local-host; local-net; };
```

- Authoritative Master

```
options {
        recursion no;
        dnssec-enable yes;
        dnssec-validation no;
};
zone "ex.de." IN {
        type master; file "ex.de.signed";
        allow-transfer {
                local-host; key "slave1-2016"; key "slave2-2016";
        };
};
```

BIND Dynamic Updates



- Signieren des neuen Eintrags
  Zugriff auf Schlüsselmaterial notwendig / Erhöhen der Seriennummer
- Regelmäßiges Resigning (Verteilt über Resigning Intervall)

```
auto-dnssec maintain;
sig-validity-interval 10 7;
```

BIND Dynamic Updates (2)

- Pro
  - + Kein Handling der Seriennummer nötig
  - + Nur noch Veränderungen senden, nicht mehr vollständige Zone
  - + Incrementelle Zonnentransfer (IXFR)
  - + Unterschiedliche Quellen für Zoneninhalt möglich
- Contra
  - − Große Anpassungen an Provisionierungsdienst
  - − Kein „vi“-Mode möglich (siehe `zkt-ddns`)
  - − Authentifizierung notwendig (SIG0, TSIG, GSS-TSIG)
- Config

```
zone "ex.de." in {
        type master; file "ex.de.dyn";
        key-directory "/var/named/keys";
        sig-validity-interval 30;
        dnssec-update-mode maintain;    // re-sign all records not just new ones
        serial-update-method increment; // increment or unxitime (or date)
        update-policy {
                grant ddns-key zonesub any;
                grant * self * A AAAA TXT SSHFP SRV HIP;
        };
};
```

BIND Smart Signing

- Schlüssel bekommen Timerwerte
  - Created: Offensichtlich der Zeitpunkt der Schlüsselerzeugung
  - Publish: Zu diesem Zeitpunkt DNSKEY Record in die Zone aufnehmen
  - Activate: Ab jetzt zum Signieren verwenden
  - Inactive: Ab jetzt nicht mehr zum Signieren verwenden
  - Delete: Aus der Zone entfernen
- Für Key Signing Keys zusätzlich
  - Revocation: Zurückziehen eines KSK
  - SyncPublish: Zeitpunkt der Bekanntmachung als CDS/CDNSKEY
  - SyncDelete: Entfernen des CDS/CDNSKEY Record
- Kommando zum Setzen der Timer

```
dnssec-settime
    -P date/[+-]offset/none: set/unset key publication date
    -A date/[+-]offset/none: set/unset key activation date
    -R date/[+-]offset/none: set/unset key revocation date
    -I date/[+-]offset/none: set/unset key inactivation date
    -D date/[+-]offset/none: set/unset key deletion date
```

BIND Smart Signing

- Anzeigen der Timerwerte eines Keys

```
$ grep "^; " Kex.de.+005+49618.key
; This is a key-signing key, keyid 49618, for ex.de.
; Created: 20160430165432 (Sat Apr 30 18:54:32 2016)
; Publish: 20160430165432 (Sat Apr 30 18:54:32 2016)
; Activate: 20160430165432 (Sat Apr 30 18:54:32 2016)
; SyncPublish: 20160430165432 (Sat Apr 30 18:54:32 2016)
```

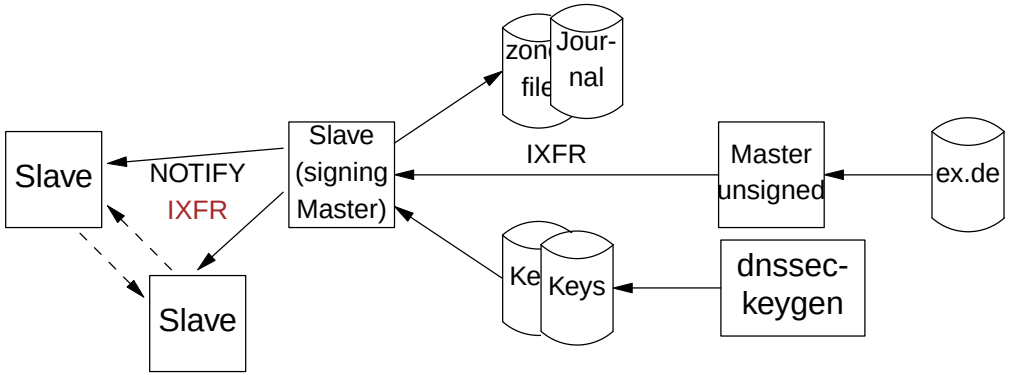
```
$ dnssec-settime -p all Kex.de.+005+49618.
```

- Für mehrere Schlüssel

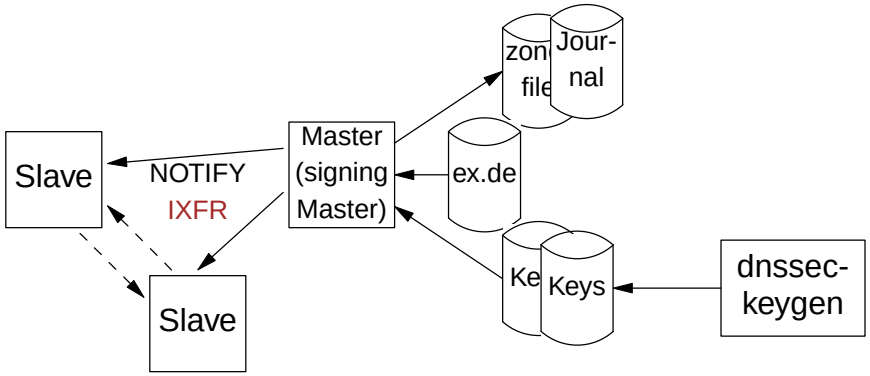
```
$  zkt2-ls -L -w10 -a .
Keyname      Tag Typ Sta Algorit crea publ acti inac dele revo cpub cdel
ex.de.     01837 KSK act RSASHA1 <4y  <4y  <4y
ex.de.     49618 KSK act RSASHA1 <6w  <6w  <6w                  <6w
ex.de.     31146 ZSK cre RSASHA1 <6w  >40w >45w
ex.de.     23085 ZSK cre RSASHA1 <6w  >15w >19w >45w >47w
ex.de.     53883 ZSK act RSASHA1 <6w  <6w  <6w  >19w >21w
ex.de.     18222 ZSK ina RSASHA1 <12w <12w <11w <6w
ex.de.     52740 ZSK del RSASHA1 <4y  <4y  <4y  <11w <11w
```

BIND Inline Signing

- Signing Feature dynamischer Zonen verbunden mit statischen Zonen
  - Input kommt über Zonentransfer (Inline Signing Slave Server)



  - oder aus statischem Zonenfile (Inline Signing Master Server)



BIND Inline Signing (2)

- Config

```
zone "ex.de" in {
    type master;   /* oder slave */
    file "ex.de/zone.db";
    inline-signing yes;
    auto-dnssec maintain;
    serial-update-method increment;
    sig-validity-interval 10 7;
};
```

- Bedeutet die Verwaltung zweier Zonen ...
  - Erhöhten Speicherbedarf beachten
  - Zwei Zonendateien (Vier mit Journal Files)

```
$ ls zone.db*
zone.db  zone.db.jnl  zone.db.signed  zone.db.signed.jnl
```

  - Unterschiedliche Seriennnummer (!)

```
$ rndc zonestatus ex.de | grep serial
serial: 2016032201
signed serial: 2016032276
```

  - Im Zweifel änderbar über nicht dokumentiertes Kommando

```
$ rndc signing -serial 2016061501 ex.de
```

DNSSEC Policy Empfehlung

- Immer zwei Schlüssel (KSK/ZSK) nutzen
  - Zone Signig Key signiert „alle“ Records
  - Key Signing Key signiert ausschließlich DNSKEY Records
  - ZSK **nicht** zum Signieren der DNSKEY Records benutzen (`ksk-only yes;` bzw. `dnssec-signzone -x`)
- Key Signing Key
  - RSASHA256; 4096 (Bind def.: 2048)
  - Double-Signing Rollover alle 2 − 5 Jahre
  - Nur einmalige Änderung des DS Records beim Parent
- ZSK Rollover
  - RSASHA256; 2048 (Bind def.: 1024)
  - Pre-Publish Rollover alle 3 bis 12 Monate
  - Hält Zonengröße konstant da nur eine Signatur
  - Pre-Publish unterstützt von `dnssey-keymgr`
- Zone Resigning
  Alle 10 bis 30 Tage (Bind default: 30 Tage)

DNSSEC Policy Empfehlung

- Zwei Arten von „Nichts“
  Wie signiert man nicht vorhandene Einträge?
- NSEC
  - Standardverfahren; Erlaubt Enumeration einer Zone
- NSEC3 (RFC5155)
  - Benutzt Hashwerte statt Label
  - Mehraufwand beim Validieren
  - DENIC benutzt NSEC3 für die `.de` Zone

    ```
    $ dig +short nsec3param de
    1 0 15 BA5EBA11
    ```
  - Nebeneffekt: Opt-out Signing
- Wenn’s denn unbedingt sein muß:
  - Hash Algorithmus: SHA1
  - Anzahl Iterationen: 10 bis 15
  - Salt: Zufällig, 8 Hex Ziffern

```
$ rndc signing -nsec3param 1 0 10 03F92714 ex.de
```

Questions ?

DNSSEC, VoIPsec, IPsec, XMPPsec, WLANsec, ...

... DKIM, Kerberos, Radius, NTP, DHCP, DNS, ...

... IPv6, Routing, Switching, SDN

Holger.Zuleger@hznet.de

Dynamische Zonen

- Internet entwickelt sich hin zu dynamischer Nutzung
  - Mobile Geräte (Laptop / Smartphone)
  - Wireless Netzzugänge (WLAN, 3/4G, Bluetooth)
  - IPv6 Adresszuweisung erlaubt automatisiertes Renumbering
- Klasssiche DNS stammt aus einer anderen Zeit
  - Server, Desktop PCs, Ethernet
  - Zonenfile anlegen, konfigurieren, testen, ... und danach vergessen
  - Änderungen selten notwendig
- Auch DNSSEC erhöht Anforderungen bzgl. Zonenänderungen
  Resigning der Zone oder einzelner Resource Records
- Inkrementelle Zonentransfer (IXFR) nutzen
  - Automatisch bei dynamischen Zonen
  - Für statische Zonen über `ixfr-from-differences yes`
  - Journal Dateien sind im Binärformat

    Anzeigen mit `named-journalprint zone.db.jnl` oder
    `named-checkzone -D -J zone.db.jnl ex.de. zone.db`

Dynamische Zonen

- Mix von statischen und dynamischen Einträgen

```
zone dyn.de. {
        type master; file "zone.dyn"; }
        update-policy {
                grant local-ddns zonesub any;    // dns admin darf alles
                grant * self * A AAAA TXT SSHFP SRV HIP;
        };
};
```

- Admin nutzt immer nsupdate (auch für „statischen“ Content)

```
{
    echo "update del ns1.ex.de. A"
    echo "update add ns1.ex.de. 86400 A 192.0.2.53"
    echo "send"
} | nsupdate -l
```

- Oder Tony Finch’s `nsdiff` Tool (http://dotat.at/prog/nsdiff/nsdiff.html)
  - Editieren eines Zonenfiles wie zuvor
  - Anstelle eines rndc reload wird nsdiff Kommando genutzt
  - Holt aktuelle Zone per AXFR
  - Erzeugt diff zum Zonenfile und sendet diesen per nsupate in die Zone

```
$ nsdiff ex.de. zone.dyn
```

Dynamische Zonen

- Mit `zkt-diff` wird Zonendatei wie bisher erstellt

```
$ORIGIN dyn.de.
$TTL 86400  ; 1 day
@              IN SOA  ns1.example.net. hostmaster.example.net. (
                       666 43200 1800 604800 900 )
               IN NS   ns1.example.net.

ns1       3600 IN A    192.0.2.53
               IN AAAA 2001:db8:18::53

-www           IN AAAA 2001:db8:af:5::3f5
+www           IN AAAA 2001:db8:af:18::3f5
               IN A    192.0.2.5
```

- Neuzugänge mit '+' am Anfang der Zeile markieren; Löschungen mit '–'

```
$ zkt-ddns -n zone.dyn
update del www.dyn.de.   86400     IN   AAAA 2001:db8:af:5::3f5
update add www.dyn.de.   86400     IN   AAAA 2001:db8:af:18::3f5
send
```

- Option `-w` entfernt diese Zeichen und schreibt Zonendatei zurück

```
$ zkt-ddns -w -n zone.dyn | nsupdate -l
$ grep -C3 ^www"zone.dyn
;www          IN AAAA 2001:db8:18:5::3f5
www           IN AAAA 2001:db8:af:18::3f5
              IN A    192.0.2.5
```

Ausblick: Dynamisch Zonen anlegen

- Dynamisches Anlegen/Entfernen von Zonen
  - Realisiert über Control Channel
  - Setzt globale Konfig `allow-new-zones yes;` voraus
  - rndc addzone/modzone <zone> "{ *Zoneconfig in BIND Syntax* };"
  - rndc delzone <zone>
  - rndc showzone <zone>
- Voraussetzung für Catalog Zones
  - Automatisierte Konfiguration von Zonen auf Slave Servern
  - Spezielle Zone enthält Infos über Zonen die verteilt werden
  - Slaves als Slave für Catalog Zone einrichten...
    ... und als Catalog Zone konfigurieren

```
catalog-zones {
     zone "catalog.example";
     default-masters { 10.53.0.1; };
     zone-directory "catzones";
     min-update-interval 10;
     };
```

  - Anlegen/Löschen/Modifizieren von Zonen nur noch auf Master notwendig

Ausblick: Keyrollover

- Neues Tool zur (Pre) Generierung von Nachfolgeschlüssel
  - Aktuell nur automatisierter (ZSK) Rollover
  - Pre-Publish Key Rollover
  - KSK Rollover vorbereitet und geplant
  - Python Skript `dnssec-keymgr`
- Nutzt neue Policy Datei

```
$ cat /etc/policy.conf
policy default {
    policy global;
    algorithm nsec3rsasha1;
    key-size zsk 1024;
    pre-publish zsk 6w;
    post-publish zsk 6w;
    roll-period zsk 6mo;
    roll-period ksk 0;
    coverage 364d;
};
```

- Überprüfung ob Schlüssel lückenlos sind
  Python Programm `dnssec-coverage`

Managed Keys Management

- Anzeige und Verwaltung der Mangaged Keys
  - Mangaged Keys sind Trust Anchor die RFC5011 Rollover nutzen
  - Auf validating Resolver konfiguriert
  - Wichtig für ein Wechsel des Root Key ohne manuelle Umkonfiguration
- Managed Keys gibt es bereits seit vielen Jahren

```
managed-keys {
    "."  initial-key  257 3 8
                 "AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQbSEW0O8gcCjF
                          ...
                 QxA+Uk1ihz0=" ; # key id = 19036
    "secsip.de."  initial-key 257 3 5
                 "AwEAAQ0tVSlTD7UPr0YQuWpWYNBxt/NyJX61UoLNbsE2
                       ...
                 WKE=" ; # key id = 22801
};
```

- Welcher Key wird tatsächlich genutzt ?

```
$ dig +noall +answer +nottl +noclass +nocrypto +dnssec DNSKEY secsip.de
secsip.de.  DNSKEY  257 3 5 [key id = 60988]
secsip.de.  DNSKEY  257 3 5 [key id = 13216]
secsip.de.  DNSKEY  256 3 5 [key id = 27453]
secsip.de.  DNSKEY  256 3 5 [key id = 26808]
secsip.de.  RRSIG   DNSKEY 5 2 7200 20160519041933 20160509041933 13216 secsip.de. [om
```

Managed Keys Management

- Neu sind die Admintools

```
$ rndc managed-keys status
next scheduled event: Wed, 13 May 2016 10:10:07 GMT

    name: .
    keyid: 19036
        algorithm: RSASHA256
        flags: SEP
        next refresh: Thu, 14 May 2016 09:10:07 GMT
        trusted since: Sat, 05 Jan 2013 14:49:27 GMT

    name: secsip.de
    keyid: 13216
        algorithm: RSASHA1
        flags: SEP
        next refresh: Wed, 13 May 2016 10:10:07 GMT
        trusted since: Mon, 18 Jan 2016 09:46:09 GMT
    keyid: 60988
        algorithm: RSASHA1
        flags: SEP
        next refresh: Wed, 13 May 2016 10:10:07 GMT
        trusted since: Tue, 14 Apr 2016 18:15:08 GMT
```

- Erneuerung:

  `rndc managed-keys refresh`

- Aktuelle Keys in Journal Datei speichern (managed-keys.bind.jnl)

  `rndc managed-keys sync`

Tipps & Tricks / Vermischtes

- Anzeige des Status einer Zone

```
$ rndc zonestatus ex.de
name: ex.de
type: master
files: z/ex.de./zone.db
serial: 2016061401
signed serial: 2016061403
nodes: 6
last loaded: Sun, 12 Jun 2016 19:54:50 GMT
secure: yes
inline signing: yes
key maintenance: automatic
next key event: Tue, 14 Jun 2016 13:14:01 GMT
next resign node: ns1.ex.de/NSEC
next resign time: Tue, 14 Jun 2016 18:48:44 GMT
dynamic: no
reconfigurable via modzone: no
```

- Anzeige der Konfiguration einer Zone

```
$ rndc showzone ex.de
zone "ex.de" in { type master; file "ex.de./zone.db";
auto-dnssec maintain; dnssec-dnskey-kskonly yes;
inline-signing yes; key-directory "ex.de.";
serial-update-method date; sig-validity-interval 10 7; };
```

Tipps & Tricks / Vermischtes

- Neue Methode zur Bildung der Seriennummer wenn BIND signiert

  `serial-update-method increment|unixtime|date;`

  - Increment: Immer um eins erhöhen
  - Unixtime: Zeit in Sekunden seit 1.1.1970
  - Date: YYYYmmddNR z.B. 2016061501

- Inline (signing) Dateien sind immmer im RAW Format
  - (Raw) Zonendatei anzeigen (Mit journal file)

    ```
    named-checkzone -D -j -s relative -f raw ex.de. zone.db.signed
    ```

  - Mit resign Zeitstempel

    ```
    named-checkzone -D -j -s full -f raw ex.de. zone.db.signed | \
    pr -e4 -l1 | cut -c1-90
    ```

- Auch für slave Dateien ist RAW Format der Standard
  Umstellen mit `masterfile-format text;`

Links

- Zone Key Tool
  - Git: `https://github.com/hzuleger/ZKT`
  - BIND tar file: Im contrib Verzeichnis (`contrib/zkt-1.1.3`)
  - Webseite: `http://www.zone-key-tool.de`
- Alpha Versionen von `zkt2-ls` und `zkt-ddns`
  `http://www.hznet.de/tools.html#zkt2`
- Blog zu Catalog Zones von J.P. Mens
  `http://jpmens.net/2016/05/24/catalog-zones-are-coming-to-bind-9-11/`
- BIND Adminstrator Reference Manual (latest Version)
  `http://ftp.isc.org/isc/bind9/9.11.0a3/doc/arm/Bv9ARM.html`
- Blog zum gleichen Thema von Daniel Stirnimann
  „DNSSEC signing your domain with BIND inline signing“
  `https://securityblog.switch.ch/2014/11/13/dnssec-signing-your-domain-with-bind-inline-signing/`

Questions ?

DNSSEC, VoIPsec, IPsec, XMPPsec, WLANsec, ...

... DKIM, Kerberos, Radius, NTP, DHCP, DNS, ...

... IPv6, Routing, Switching, SDN

Holger.Zuleger@hznet.de

CONTENTS