

1. Agenda und Folien
2. Allgemeine Informationen
3. DNS-Server VMs
4. Hostnamen, Benutzer und Passwörter
5. DNSSEC
6. DNSSEC Monitoring

DNSSEC Workshop @ DE-CIX MeetingCenter (eco)

1 Agenda und Folien

1.1 DNS & DNSSEC 14.09.2023

- 10:30 Begrüssung
- 10:45 DNSSEC Anwendungen "Securing-Security" (./dnssec-dane-tlsa.html)
- 11:30 Pause
- 11:45 Das kleine DNS 1x1 (./dns-1x1.html)
- 12:15 Praxis: DNS Resolver einrichten
- 12:30 Mittagspause
- 13:30 Einführung in DNSSEC (./dnssec-intro.html)
- 14:30 Praxis: DNS Server und DNS Zone einrichten
- 15:00 Pause
- 15:15 DNSSEC mit BIND 9 (./dnssec-sign-bind9.html)
- 16:15 DNSSEC Key-and-Signing Policy
- 17:00 Ende Workshop Tag 1

1.2 DNS & DNSSEC 15.09.2023

- 09:00 DNSSEC Schlüsselparameter (./dnssec-keyparameter.html)
- 10:00 NSEC oder NSEC3? (./nsec-vs-nsec3.html)
- 10:30 Pause
- 10:45 DNSSEC Betrieb
- 12:30 Mittagspause
- 13:15 DNSSEC Key-Rollover (./keyrollover.html)
- 14:30 Erweiterte DNS/DNSSEC Fehler-Informationen
- 15:00 Ende Workshop Tag 2

2 Allgemeine Informationen

- Diese Anleitung und die Folien finden Sie online im Internet unter <https://dnssec.dane.onl> (<https://dnssec.dane.onl>)
- Diese Webseite wird im Kurs regelmässig aktualisiert. Wenn Informationen hier fehlen, dann bitte die Webseite im Browser neu laden (reload) um die neuen Informationen abzurufen.
- In den Anleitungen müssen Sie die Zeichenfolge XX durch die zweistellige Teilnehmer-Nummer ersetzen. Die Teilnehmer-Nummer finden Sie in der Tabelle der Teilnehmer in diesem Dokument.
- Jeder Teilnehmer bekommt zwei Linux-Maschinen im Internet (ein primärer DNS-Server und ein DNS Resolver). Es ist je ein aktuelles Debian 12 installiert. Diese Linux-Maschinen ist per SSH (Secure-Shell) und per Web-Browser (Cockpit-Terminal) erreichbar.

3 DNS-Server VMs

- Ihr DNS-Server (autoritativ): `dns.zXX.dane.onl`
- Ihr DNS-Resolver: `dnsr.zXX.dane.onl`
- Text-Editoren auf den Servern: `emacs`, `vim`, `nano`, `mg` (MicroEmacs)
- Hilfsprogramme: `tmux`, `dvtm`
- Bash-Shell muss manuell per `bash` gestartet werden

4 Hostnamen, Benutzer und Passwörter

Login via SSH (OpenSSH, Putty, Google Chrome mit SSH app) oder per Browser auf Port 443 (HTTPS) des jeweiligen Servers (z.B. <https://dnsr.z01.dane.onl> (<https://dnsr.z01.dane.onl>)). Die Cockpit-Oberfläche ist mit einem selbst-signierten TLS Zertifikat abgesichert, es gibt daher die Warnung das diese Verbindung nicht "sicher" ist (der Server nicht authentisiert werden kann). In der Cockpit Web-Oberfläche auf der linken Seite das Terminal auswählen (die anderen Funktionen des Programms "Cockpit" werden wir nicht nutzen).

- Benutzername für die VMs: `user`
- Password: `dnssec-2023`
- Root-Shell mit `sudo -s` und dem Benutzer-Passwort

Teilnehmer Nr.	Name	DNS Resolver	Autoritativer Server
01		https://dnsr.z01.dane.onl (https://dnsr.z01.dane.onl)	https://dns.z01.dane.onl (https://dns.z01.dane.onl)

Teilnehmer Nr.	Name	DNS Resolver	Autoritativer Server
02		https://dnshr.z02.dane.onl (https://dnshr.z02.dane.onl)	https://dns.z02.dane.onl (https://dns.z02.dane.onl)
03		https://dnshr.z03.dane.onl (https://dnshr.z03.dane.onl)	https://dns.z03.dane.onl (https://dns.z03.dane.onl)
04		https://dnshr.z04.dane.onl (https://dnshr.z04.dane.onl)	https://dns.z04.dane.onl (https://dns.z04.dane.onl)
05		https://dnshr.z05.dane.onl (https://dnshr.z05.dane.onl)	https://dns.z05.dane.onl (https://dns.z05.dane.onl)
06		https://dnshr.z06.dane.onl (https://dnshr.z06.dane.onl)	https://dns.z06.dane.onl (https://dns.z06.dane.onl)
07		https://dnshr.z07.dane.onl (https://dnshr.z07.dane.onl)	https://dns.z07.dane.onl (https://dns.z07.dane.onl)
08		https://dnshr.z08.dane.onl (https://dnshr.z08.dane.onl)	https://dns.z08.dane.onl (https://dns.z08.dane.onl)

5 DNSSEC

5.1 Praxis: DNSSEC Signaturen

- Arbeit auf dem DNS-Resolver-Rechner
- Frage nach der DNSSEC-Signatur von `dnssec.works NS` :

```
$ dig @9.9.9.9 dnssec.works NS +dnssec +multi
```

- Beantworten Sie diese Fragen
 - Welchen Algorithmus benutzt diese Zone? Finde die Algorithmus Nummer in IANA Domain Name System Security (DNSSEC) Algorithm Numbers (<https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xhtml>)
 - Wann läuft die Signatur ab?
 - Wann ist die Signatur gültig geworden?

5.2 Praxis: DNSSEC Resolver installieren

- Installiere die DNS-Abfrage-Tools (`nslookup` , `dig`)

```
% apt install dnsutils
```

- Installiere den BIND 9 DNS Server (DNSSEC Validierung ist standardmässig angeschaltet, keine Konfiguration notwendig)

```
% apt install bind9
```

- Teste DNSSEC Validierung (AD-Flag muss angezeigt werden)

```
# dig @localhost eco.de +multi

; <<>> DiG 9.18.18 <<>> eco.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61985
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 2bd0f7a41860386e010000006502a25e783292436f13a567 (good)
;; QUESTION SECTION:
;eco.de.                        IN      A

;; ANSWER SECTION:
eco.de.                        86400   IN      A      46.31.121.137

;; Query time: 3799 msec
;; SERVER: 100.64.1.254#53(100.64.1.254) (UDP)
;; WHEN: Thu Sep 14 08:04:14 CEST 2023
;; MSG SIZE rcvd: 79
```

5.2.1 DNS/DNSSEC Flags in DNS-Paketen

Flag	Beschreibung
AA	Authoritative Answer - Antwort kommt von einem DNS Server, welcher der Besitzer der Daten ist
TC	Truncated - DNS Antwort passt nicht in die per EDNS ausgehandelte maximale UDP Paketgrösse
RD	Recursion Desired - DNS Client braucht eine finale Antwort - Zwischenantworten sind nicht erwünscht
RA	Recursion Available - DNS Resolver bietet finale Antworten (= dies ist ein Resolver, kein autoritativer Server)
Z	Zero - kann nicht benutzt werden

Flag	Beschreibung
AD	Authentic Data - DNSSEC Daten wurden vom DNS Resolver erfolgreich geprüft
CD	Checking Disabled - Für diese Anfrage die DNSSEC Prüfung ausschalten

5.3 Praxis: DNS Server und DNS Zone einrichten

- Diese Anleitung befinden sich unter <https://dnssec.dane.onl> (<https://dnssec.dane.onl>)
- Die virtuelle Labormaschine verwenden (Autoritativer DNS Server):
`dns.zXX.dane.onl`
- Als *root*-Benutzer arbeiten

```
$ sudo -s
```

- BIND 9 auf dem autoritativen DNS Server installieren

```
% apt install bind9
```

- Prüfe das der BIND 9 Prozess gestartet ist

```
% systemctl status bind9
```

- Öffne die BIND 9 Konfigurationsdatei `/etc/named.conf` mit einem Text Editor. Eine primäre autoritative DNS-Zone für die Domain `zXX.dane.onl` ist schon konfiguriert

```
% chmod +w /etc/bind/named.conf
% nano /etc/bind/named.conf
```

- Die Zonendatei `/etc/bind/zonefile.db` schreibbar markieren (`chmod +w /etc/bind/zonefile.db`) und diese Datei in einem Editor öffnen. Füge im Apex der Zone (für den Basis-Domain-Namen der Zone) einen IPv6 AAAA-Record für die IPv6-Adresse des Servers hinzu. Die IPv6-Adresse lässt sich unter anderem mit dem Befehl `hostname -I` anzeigen. Erhöhe die SOA-Seriennummer und speichere die Änderungen.
 - Beispiel AAAA-Record:

```
zXX.dane.onl.      IN AAAA 2001:db8::1
```

- Prüfen Sie die BIND 9-Konfigurationsdatei und die Zonen-Datei auf Fehler

```
% named-checkconf -z
```

- Wenn keine Fehler gemeldet werden, lade die neue Zone erneut in den BIND 9 Server (über rndc)

```
% rndc reload zXX.dane.onl
```

- Test, ob Ihr DNS-Server auf Anfragen für diese Zone antwortet und die neue SOA-Seriennummer zeigt

```
$ dig @::1 zXX.dane.onl SOA
```

- Verwende einen externen DNS-Resolver (in diesem Beispiel Quad9) oder den eigenen DNS-Resolver auf dnsr.zXX.dane.onl, um die IPv6 Adresse aus der DNS-Zone aufzulösen

```
$ dig @9.9.9.9 zXX.dane.onl AAAA
```

5.4 Praxis: Einfaches DNSSEC mit BIND 9.16 "default-policy"

- Die Verwendung der DNSSEC Policy-Konfiguration (verfügbar seit BIND 9.16) ermöglicht die einfache Implementierung von DNSSEC
- BIND erzeugt automatisch die DNSSEC-Schlüssel, signiert die Zone und hält die Signaturen aktuell
- Die Direktive `default` bewirkt, dass die Zone mit einem einzigen kombinierten Signierschlüssel (CSK) mit dem Algorithmus ECDSA256SHA256 signiert wird. Dieser Schlüssel hat eine unbegrenzte Lebensdauer (kein Schlüssel-Rollover).
- Wir arbeiten auf dem primären autoritativen Server
- Bearbeite die Datei `/etc/bind/named.conf` in einem Editor. Füge in der Zonen-Definition die Zeile `dnssec-policy default;` und `inline-signing yes;` der Zonen-Definition hinzu:

```
zone "zXX.dane.onl" {
    type primary;
    file "zonefile.db";
    dnssec-policy default;
    inline-signing yes;
};
```

- Prüfe die Konfiguration und starte den BIND 9 Server neu

```
% named-checkconf -z
% systemctl start bind9
```

- Die Schlüssel-Dateien und weitere Zonen-Dateien sollen nun im BIND 9 Heimverzeichnis sichtbar werden

```
% ls -l /etc/bind
% ls -l /etc/bind/keys
```

- Erstelle einen neuen DS Record mit `dnssec-dsfromkey` und speicher den DS-Record als Text-Datei unter `/etc/bind/ds-record.txt`. Der Trainer wird den DS-Record dort finden und in der Eltern-Zone veröffentlichen.

```
dnssec-dsfromkey /etc/bind/keys/Kz08.dane.onl.+013+YYYYY.key # \
# S
```

- Teste das der DNS-Server die DNSSEC Resource Records zurückliefert

```
$ dig @localhost zXX.dane.onl +dnssec +multi
$ dig @localhost zXX.dane.onl DNSKEY +dnssec +multi
```

- Frage mit `dig` den A-Record von `zXX.dane.onl` über einen externen DNS-Resolver an. Wird das AD-Flag angezeigt?

5.5 DNSSEC KASP

- BIND 9.16 führt eine neue Funktion mit dem Namen `dnssec-policy` ein, die einen weiteren Schritt in der Automatisierung über `auto-dnssec maintain` hinaus anbietet
 - Es ist wahrscheinlich, dass `dnssec-policy inline signing` ersetzt wird (daher die Warnungen in neuen BIND 9 Versionen das `auto-dnssec maintain abgekündigt` (deprecated) ist)
- Wo *inline* oder *dynamic* erwarten, dass die Schlüssel mit `dnssec-keygen` erstellt wurden, automatisiert diese Signiermethode die Aufgabe
- Die Konfiguration einer Zone für die Verwendung von KASP (*Key And Signing Policy*) kann so einfach sein wie

```
zone "example.net" {
    type primary;
    dnssec-policy default;
    ...
};
```

- Vorteile
 - Intuitiv und ein höheres Maß an Automatisierung
 - Mehrere Anbieter verwenden KASP Systeme (Knot, OpenDNSSEC)
 - Robust
 - Keine Notwendigkeit, sich auf von Menschen hinzugefügte Metadaten zu verlassen
 - Verwendung einer Key-Timing State-Machine

5.5.1 Standard BIND 9 DNSSEC Richtlinie (KASP)

- Einzelner CSK
 - ECDSAP256SHA256 (Algo 13) mit unbegrenzter Lebensdauer
 - RRSIG Gültigkeit 14 Tage, aufgefrischt 5 Tage vor Ablauf
- NSEC
- Schlüssel-Timings:
 - DNSKEY TTL: 3600, maximale Zonen-TTL: 86400 (1 Tag)
 - Sicherheitszeiten für die Veröffentlichung und das Zurückziehen von Schlüsseln: 3600 (1 Stunde)
 - Ausbreitungsverzögerung: 300 (5 Minuten)
- Zeitvorgaben für die Eltern-Zone
 - DS TTL: 86400 (1 Tag)
 - Ausbreitungsverzögerung: 3600 (1 Stunde)
- Durch die Verwendung einer explizit definierten DNSSEC-Richtlinie anstelle der default Richtlinie verhindert die ungewollte Änderung nach Updates der BIND 9 Software

5.5.2 Eigene KASP Policy

```
dnssec-policy "mypolicy" {
    keys {
        ksk lifetime 365d algorithm rsasha256 2560;    // nur Beispiel
        zsk lifetime 60d  algorithm rsasha256 1536;    // nicht in F
        csk lifetime P6MT12H3M15S algorithm 13;        // verwenden
    };
};

zone "example.net" {
    dnssec-policy "mypolicy";
    [...]
};
```

- `keys` gibt *Rollen* anstelle von bestimmten Schlüsseln an
- `lifetime` gibt die Dauer oder unbegrenzt an
- `algorithm` verwendet Mnemonik oder Zahl

5.5.3 Zusätzliche Konfiguration der benutzerdefinierten Richtlinie


```

dnssec-policy "one" {
    keys {
        ksk lifetime 365d algorithm rsasha256 4096;
        zsk lifetime 60d  algorithm rsasha256 1024;
    };
    dnskey-ttl 600;
    publish-safety PT2H;
    signatures-refresh 7d;

    nsec3param iterations 0 optout no salt-length 0;
};

```

5.5.4 Key-and-Signing-Policy (KASP) Syntax

```

dnssec-policy <string> {
    dnskey-ttl <duration>;
    keys { ( csk | ksk | zsk ) [ ( key-directory ) ]
        lifetime <duration_or_unlimited> algorithm <string> [ <integer> ]
    };
    max-zone-ttl <duration>;
    nsec3param [ iterations <integer> ] [ optout <boolean> ] [
    parent-ds-ttl <duration>;
    parent-propagation-delay <duration>;
    publish-safety <duration>;
    purge-keys <duration>;
    retire-safety <duration>;
    signatures-refresh <duration>;
    signatures-validity <duration>;
    signatures-validity-dnskey <duration>;
    zone-propagation-delay <duration>;
};

```

5.6 DNSSEC Fehlersuche

5.6.1 Überprüfung von DNS-Auflösungsproblemen

1. dig

Das DNS-Namensauflösungstool `dig` kann verwendet werden, um die allgemeine Funktion eines DNS-Auflösers zu testen, oder um zu prüfen, ob ein Fehlerzustand bei den entfernten autoritativen DNS-Servern einer Domäne.

1. Test eines DNS-Resolvers über UDP

Um die allgemeine Konnektivität und Funktion eines DNS-Resolvers zu testen, kann eine Abfrage nach bekannten DNS-Daten gesendet werden, z. B. nach der Liste der Nameserver (NS-Records) der Root-Zone `."`.

Die Antwort sollte die Liste aller 13 Root-Nameserver im Internet (mit den Namen `a-m.root-servers.net`) enthalten

```
$ dig @IP-of-DNS-resolver NS .
```

- Was sind die Gründe, warum wir empfehlen, die IP-Adresse eines Resolvers anstelle seines Namen zu verwenden? Schreibe die Antwort in den Chat.

2. Testen eines DNS-Resolvers über TCP

Die DNS-Resolver müssen auch über TCP erreichbar sein. Um eine Anfrage über TCP zu senden, füge das `+tcp` Flag den Abfragen hinzu.

```
$ dig @IP-of-DNS-resolver NS . +tcp
```

- Sie können 33% Tipparbeit sparen, indem Sie `+vc` statt `+tcp` verwenden. Was bewirkt `vc`? Schauen Sie sich die Handbuchseite (man-page) für `dig(1)` an. Schreiben Sie die Antworten in den Chat.

3. Testen der Erreichbarkeit aller autoritativen DNS-Server

Die Funktion `dig +nssearch` fragt zuerst alle NS-Einträge für eine gegebene Domain ab und versucht dann, direkt (ohne Umweg über einen DNS-Resolver) die autoritativen DNS-Server dieser Domäne abzufragen:

```
$ dig @IP-of-DNS-resolver example.com +nssearch
```

Die Funktion gibt den SOA-Eintrag der Zone aus (hier `example.com`) für jeden DNS-Server aus, der eine Antwort auf die Anfrage gesendet hat, einschließlich der SOA-Seriennummer, der IPv4- und/oder IPv6-Adresse und der Round-Trip-Time (RTT) der Abfrage.

Alle SOA-Seriennummern sollten dieselbe Nummer aufweisen, sonst könnte es ein Problem mit der Zonensynchronisierung über den Zonentransfer geben, was eine mögliche Ursache für DNS-Lookup-Probleme sein könnte.

- welche Art von DNS-Servern (autoritativ, rekursiv, primär, sekundär) sind schuld, wenn die SOA-Seriennummern einer Zone nicht synchron sind? Wer kann den Fehler beheben? Schreiben Sie die Antworten in den Chat.

1. Testen der Auflösungskette

Die Funktion `+trace` in `dig` verfolgt die DNS-Namensauflösung vom Root-DNS-Server-System bis zum angeforderten Namen.

```
$ dig @IP-of-DNS-resolver example.com +trace
```

Nur die allererste Abfrage zur Ermittlung der Root-Server-Adressen wird an den im Befehl angegebenen DNS-Resolver gestellt, alle anderen Abfragen werden direkt an die autoritativen DNS-Server gesendet. Diese Funktion testet und gibt einen von normalerweise vielen möglichen DNS-Auflösungspfaden aus. Ein erfolgreicher Return-Code des Befehls bedeutet nicht, dass der Auflösungspfad fehlerfrei ist, sondern nur, dass mindestens ein erfolgreicher Pfad existiert.

4. Prüfung auf DNSSEC-Validierungsprobleme

Wenn eine DNS-Abfrage eine SERVFAIL -Antwort zurückgibt, kann es sich um ein DNSSEC Validierungsproblem beim DNS-Resolver handeln, oder es kann sich um eine Art Server-Fehlfunktion auf dem DNS-Resolver oder dem entfernten autoritativen Server sein.

Um nach DNSSEC-Validierungsproblemen zu suchen, kann der Administrator eine DNS-Abfrage mit dem Flag `+cd` (Checking Disabled) senden. Mit diesem Flag überspringt der DNS-Auflöser die DNSSEC-Validierung und gibt die die DNS-Daten an `dig` zurück, selbst wenn die DNSSEC-Validierung fehlschlägt.

Wenn also eine DNS-Abfrage Daten zurückgibt, wenn `+cd` gesetzt ist, aber SERVFAIL meldet, wenn `+cd` nicht gesetzt ist, deutet dies auf ein DNSSEC-Validierungs Problem. Wenn die Antwort immer SERVFAIL lautet, handelt es sich um ein anderes ein anderes Problem (normalerweise nicht DNSSEC).

```
dig @localhost fail03.dnssec.works # SERVFAIL
dig @localhost fail03.dnssec.works +cd # Antwort ohne DNSSEC
```

5.6.2 Externe Webdienste zur Überprüfung von DNS

Es gibt mehrere Webseitendienste, die DNS-Administratoren bei der Überprüfung den Zustand eines DNS-Systems helfen

1. Zonemaster

Die Website Zonemaster <https://zonemaster.net> (<https://zonemaster.net>) ist eine Zusammenarbeit zwischen der französischen TLD-Registrierungsstelle *AFNIC* und der schwedischen Registrierungsstelle *IIS*. Die Website nimmt einen Domännennamen und erstellt einen Bericht über Fehler und Best-Practice-Empfehlungen für die Einrichtung dieses Domännennamens.

2. DNSViz

DNSViz <https://dnsviz.net> (<https://dnsviz.net>) ist ein Tool zur Visualisierung des Status einer DNS-Zone. Es bietet eine visuelle Analyse der DNSSEC Authentifizierungskette für einen Domännennamen und seinen Auflösungspfad im DNS Namespace, und es listet Konfigurationsfehler auf, die das Tool entdeckt hat.

5.7 DNSSEC Probleme bei DNS Resolver Betreibern

5.7.1 Negative Trust Anchor

- DNSSEC-Validierungsprobleme werden häufig durch operative Probleme verursacht. (abgelaufene DNSSEC-Signaturen, Inkonsistenzen zwischen DS-Eintrag und DNSKEY KSK etc.)
 - Negative Trust Anchors können verwendet werden, um die DNSSEC-Validierung für fehlerhaft konfigurierte Domains zu deaktivieren
- Negative Trust Anchors sind in RFC 7646 ("Definition and Use von DNSSEC Negative Trust Anchors") definiert.
- Negative Trust Anchors (nta) deaktivieren die DNSSEC-Validierung für eine Domain für einen bestimmten Zeitraum
 - NTAs können von Betreibern verwendet werden, wenn eine fehlerhafte Konfiguration für eine DNSSEC-signierte Zone entdeckt wird. Es sollte darauf geachtet werden dass die fehlgeschlagene DNSSEC-Validierung tatsächlich von einer Fehlkonfiguration ausgelöst wird und nicht durch einen Angriff
- Domänen mit einem NTA werden so behandelt, als gäbe es keinen Vertrauensanker für diese Domain
- NTAs werden gespeichert und bleiben auch nach einem Neustart von BIND 9 erhalten.
- BIND 9 überprüft die Domäne in regelmäßigen Abständen. Sobald die Domäne wieder erfolgreich DNSSEC validiert wurde, wird die NTA für die Domain entfernt.
- NTAs haben eine begrenzte Lebensdauer (maximal eine Woche) und verfallen automatisch.
- Beispiel: Hinzufügen eines NTA (für 60 Sekunden):

```
% rndc nta -l 60 fail01.dnssec.works
Negative trust anchor added: fail01.dnssec.works/_default, expires 18
% rndc nta -dump
fail01.dnssec.works: expired 18-Aug-2016 13:52:19.000
% ls -l /var/named/_default.nta
-rw-r--r--. 1 root root 44 Aug 18 13:51 /var/named/_default.nta
% cat /var/named/_default.nta
fail01.dnssec.works. regular 20160818115219
```

- Beispiel: einen NTA entfernen

```
% rndc nta -l 86400 fail02.dnssec.works # add a NTA for 1 day
Negative trust anchor added: fail02.dnssec.works/_default, expires 19

% rndc nta -dump
fail02.dnssec.works: expiry 19-Aug-2016 13:56:22.000

% rndc nta -r fail02.dnssec.works # remove the NTA
Negative trust anchor removed: fail02.dnssec.works/_default

% rndc nta -dump      # NTA is now gone
```

5.7.2 Praxis: einen NTA für die Domain failNN.dnssec.works einrichten

- Im Kapitel zur DNSSEC-Fehlerbehebung haben wir über die defekten DNS-Zonen fail01.dnssec.works bis fail05.dnssec.works gesprochen.
- Erstellen Sie negative Trust Anchors (NTA) für fail01 bis fail03 im DNSSEC-validierenden Resolver dnssrNN
- Überprüfe, ob ein Client nun DNS-Daten aus diesen Zonen abrufen kann.

```
$ dig @127.0.0.1 fail01.dnssec.works A
```

- Entferne den NTA für fail01.dnssec.works . Prüfe das die Zonen nun nicht mehr validiert und SERVFAIL zurückliefert

5.7.3 Empfehlungen für Betrieb von DNS Resolvem

- Stelle sicher, dass der DNSSEC-validierender Resolver DNSSEC NTA (negative trust anchor) unterstützt.
- Testen Sie die Funktion des negativen Vertrauensankers.
- Dokumentieren Sie, wie ein negativer Vertrauensanker hinzugefügt wird.
- Wenn Ihre Software das automatische Entfernen von NTAs nicht unterstützt, implementieren Sie eine Automatisierung (in Verbindung mit der Überwachung Ihres DNS-Resolvers).
- Implementieren Sie keine automatische Hinzufügung von NTAs, die Aktivierung eines NTAs sollte erst nach menschlicher Überprüfung des DNSSEC-Validierungsproblems erfolgen.
- Lesen Sie den Internet-Entwurf Recommendations for DNSSEC Resolvers Operators (<https://www.ietf.org/archive/id/draft-ietf-dnsop-dnssec-validator-requirements-01.html>)

5.7.4 DNSSEC Resolver und die Systemzeit

- Die DNSSEC-Validierung benötigt eine korrekt Systemzeit.

- Die DNSSEC-Signaturen haben Ablauf- und Inception-Zeiten (*Beginn der Gültigkeit*), welche der DNS-Resolver überprüfen muss.
- Ohne eine korrekte Zeitangabe auf dem DNSSEC-Resolver-Rechner kann die DNSSEC Validierung fehlschlagen.
- Empfehlung: Implementieren Sie eine automatische Zeitsynchronisation für den DNSSEC-Resolver (NTP, PTP)
- Stellen Sie sicher, dass die Zeitsynchronisation nicht von (DNSSEC) abhängig ist (keine zirkulären Abhängigkeiten oder *Henne-Ei-Problem*).

5.8 DNSSEC Neuigkeiten

5.8.1 DNS/DNSSEC error reporting

- Der IETF-Entwurf "DNS error reporting" beschreibt eine Möglichkeit für DNS Resolver-Software, um Fehlerzustände bei der DNS-Namensauflösung an den Betreiber der autoritativen DNS-Zone zu melden.
- Diese Funktion ist vergleichbar mit dem DMARC-Reporting bei E-Mails.
- Die gemeldeten Fehler basieren auf "Extended DNS Errors (EDE, RFC 8914)".
- Die Meldung erfolgt innerhalb des DNS-Protokolls.
 - Der autoritative DNS-Server kann mit einer EDNS-Option die erweiterten Fehlercodes anfordern
 - Der autoritative DNS-Server kann wählen, ob er nur Fehlerzustände oder auch positive Rückmeldungen (bei erfolgreicher Namensauflösung) erhalten möchte.
 - Um einen Fehler zu melden, kodiert der meldende Resolver den Fehlerbericht in den *QNAME* der Anfrage. Der meldende Resolver bildet diesen *QNAME* durch Verkettung des *_er* Labels, den erweiterten Fehlercode, den *QTYPE* und den *QNAME*, der zu dem Fehler zum Fehler führte, nochmals das *_er* Label und die Domain des sendenden DNS-Resolvers
 - Der resultierende Domainname (*QNAME*) wird für die DNS-Anfrage nach einem *NULL*-Resource-Record verwendet, welcher den Fehler meldet.
 - Die *reporting agent domain* unterscheidet sich normalerweise von einer Produktionsdomäne. Die *reporting agent domain* sollte **nicht** DNSSEC signiert sein, da nur die Anfragen von Nutzen sind und die Antworten nicht gesichert werden müssen.
 - Der autoritative DNS-Server antwortet mit einer NODATA/NXRRSET Antwort, welche vom Resolver zwischengespeichert wird (die negative TTL dieser Antwort bestimmt das Intervall für Fehlerberichte, da neue Fehlerberichte für den gleichen Fehler werden erst nach Ablauf der TTL gesendet werden können).
- Beispiel (aus dem Entwurf des RFC-Draft-Dokuments):

- Die Domäne `broken.test` wird auf einer Reihe von vertrauenswürdigen Servern gehostet. Einer dieser Server bietet eine veraltete Version an. Dieser autoritative Server hat den Schweregrad 1 und einen Reporting Agent konfiguriert: `a01.reporting-agent.example`.
- Der Reporting Resolver ist nicht in der Lage, die den Resource-Record für `broken.test` zu validieren, da der RRSIG-Record eine abgelaufene Signatur enthält.
- Der meldende Resolver konstruiert den *QNAME*. `_er.7.1.broken.test._er.a01.reporting-agent.example` und löst ihn auf. Dieser *QNAME* zeigt an, dass der erweiterte DNS-Fehler 7 aufgetreten ist aufgetreten ist, als versucht wurde, den Typ 1 (A)-Record `broken.test` zu validieren.
- Nachdem diese Anfrage bei einem der autoritativen Server für die Domain (`a01.reporting-agent.example`), stellt der Empfänger des Berichts (der Betreiber des autoritativen Servers für `a01.reporting-agent.example`), dass der autoritative Server für die Zone `broken.test` unter einem abgelaufenen abgelaufenen Signatursatz (Extended Error 7) vom Typ A für den Domännennamen `broken.test` leidet. Der Melder kann den Betreiber von `broken.test` kontaktieren, um das Problem zu beheben.
- Draft "DNS error reporting" (Ein Draft ist noch kein Standard und wird es vielleicht nie werden): <https://datatracker.ietf.org/doc/draft-ietf-dnsop-dns-error-reporting/> (<https://datatracker.ietf.org/doc/draft-ietf-dnsop-dns-error-reporting/>)
- Dieser Entwurf wurde noch nicht in produktive DNS-Server-Software implementiert.

5.9 Migration BIND 9 DNSSEC zu einer KASP Policy

- BIND 9.16/9.18 führt eine neue (bessere) Automatisierung der DNSSEC-Signierung von DNS-Zonen ein: während in den älteren BIND 9 Versionen die Key-Rollover mittels der Meta-Daten auf den Schlüssel-Dateien automatisiert werden konnten, können die Key-Rollover nun innerhalb der BIND 9 Konfiguration in einer KASP (Key-and-Signing-Policy) definiert werden. Der BIND 9.18 führt ggf. Key-Rollover automatisiert durch.
- Zonen, welche schon vor BIND 9.16 oder BIND 9.18 mit DNSSEC abgesichert waren sollen auf das neue KASP System migriert werden, da die alte DNSSEC-Automatisierung in BIND 9 abgekündigt ist und in einer zukünftigen Version von BIND 9 (9.19-dev, 9.20-release) nicht mehr verfügbar sein wird.
- Für eine erfolgreiche Migration ist es erforderlich, eine DNSSEC-KASP zu definieren, welche zu den bestehenden DNSSEC-Schlüsseln in der Zone passt (Algorithmus, Anzahl Schlüssel = CSK oder KSK/ZSK).

- Erkennt der BIND 9 beim Laden der Zone ein Differenz zwischen den existierenden Schlüsseln und der neuen Policy, so wird BIND 9 die bestehenden Schlüssel **nicht** mehr benutzen und stattdessen neue Schlüssel erstellen und **sofort** die Zone mit den neuen Schlüsseln signieren. Dies kann zu einem Total-Ausfall der Zone führen, da der DS-Record in der Eltern-Zone nicht zu KSK/CSK der Zone passt.
- Bei der Migration auf eine KASP-Konfiguration sollte die Lebensdauer der Schlüssel mit "unlimited" angegeben werden. Nach der erfolgreichen Migration kann die KASP-Konfiguration angepasst und die Lebensdauer der Schlüssel begrenzt werden.
- Es ist sinnvoll, eine Migration von einer pre-KASP Konfiguration einer Zone auf eine Zone mit DNSSEC-KASP and einer Test-Zone zu üben.
- Bei einer erfolgreichen Migration auf eine KASP-Konfiguration übernimmt der BIND 9 die bestehenden DNSSEC-Schlüssel und signiert die Zone weiterhin mit diesen Schlüsseln.
- Werden neuen Schlüssel erzeugt und für die Signierung verwendet, so war die Migration nicht erfolgreich.
- Bei der Benutzung eines *hidden-primary* DNS-Servers, sollte für die Aktivierung der KASP-Konfiguration die Kommunikation zwischen den aktiven Auth-Servern und dem *hidden-primary* unterbrochen werden, bis die erfolgreiche Migration bestätigt ist (in der Regel direkt nach dem Laden der neuen Konfiguration sichtbar).
- Bei der Benutzung einer Infrastruktur ohne *hidden-primary* (der Primary ist aktiver DNS-Server in der Delegation) kann der Primary-Server temporär in einen "hidden-primary" umgewandelt werden (z.B. durch ein Block von UDP/TCP Port 53 in der Firewall). Der oder die restlichen DNS-Server werden die bestehende DNSSEC-Zone ohne Unterbrechung weiter ausliefern, bis die erfolgreiche Migration bestätigt ist und die Blockade in der Firewall aufgehoben wurde.
- Beispiel einer Pre-KASP-BIND 9 DNSSEC-Konfiguration. Die Schlüssel (ZSK/KSK) wurden manuell mit `dnssec-keygen` mit dem Algorithmus RSASHA256 erstellt:


```

options {
    directory "/var/named";
    key-directory "keys";
};

zone "example.com" IN {
    type primary;
    file "example.com";
    auto-dnssec maintain;
    inline-signing yes;
};

```

- Beispiel einer KASP BIND 9 DNSSEC-Konfiguration für diese Zone:

```

dnssec-policy "base-dnssec" {
    keys {
        ksk lifetime unlimited algorithm RSASHA256;
        zsk lifetime unlimited algorithm RSASHA256;
    };
};

options {
    directory "/var/named";
    key-directory "keys";
};

zone "example.com" IN {
    type primary;
    file "example.com";
    dnssec-policy "base-dnssec";
    inline-signing yes;
};

```

- Eine produktive KASP-Konfiguration mit ZSK-Rollover (aber keinem automatischem KSK-Rollover):

```

dnssec-policy "DNSSEC-ZSK-ROLL" {
    dnskey-ttl 60;
    keys { ksk lifetime unlimited algorithm ECDSAP256SHA256;
           zsk lifetime P30D           algorithm ECDSAP256SHA256;
    };
    max-zone-ttl 3600;
    publish-safety 1h;
    purge-keys P90D;
    retire-safety 1h;
    signatures-refresh 5d;
    signatures-validity 14d;
    signatures-validity-dnskey 14d;
    zone-propagation-delay 300;
};

zone "example.de" {
    type primary;
    inline-signing yes;
    dnssec-policy "DNSSEC-ZSK-ROLL";
    file "primary/example.de";
};

```

- Beispiel einer KASP BIND 9 DNSSEC-Konfiguration mit automatischem ZSK/KSK Rollover mittels CDS/CDNSKEY (Schweiz):

```

parental-agents "switch" {
    130.59.31.41;
    130.59.31.43;
    194.0.25.39;
    194.0.17.1;
    194.146.106.10;
    2001:620:0:ff::56;
    2001:620:0:ff::58;
    2001:678:20::39;
    2001:678:3::1;
    2001:67c:1010:2::53;
};

dnssec-policy "DNSSEC-AUTO" {
    keys {
        ksk lifetime P90D  algorithm ecdsap256sha256;
        zsk lifetime P45D  algorithm ecdsap256sha256;
    };
    max-zone-ttl 3600;
    parent-ds-ttl 600;
    parent-propagation-delay 2h;
    publish-safety 7d;
    retire-safety 7d;
    signatures-refresh 5d;
    signatures-validity 15d;
    signatures-validity-dnskey 15d;
    zone-propagation-delay 2h;
};

zone "example.ch" {
    type primary;
    allow-update { key dns-update; };
    allow-transfer { secondaries; };
    dnssec-policy "DNSSEC-AUTO";
    parental-agents {
        "switch";
    };
    file "primary/example.ch";
};

```

6 DNSSEC Monitoring

- Men & Mice DNS/DNSSEC Monitoring Scripte <https://github.com/menandmice-services/dns-monitoring-scripts> (<https://github.com/menandmice-services/dns->

monitoring-scripts)

Date: <2023-09-12 Tue>

Author: Carsten Strotmann and Patrick Koetter

Created: 2023-09-15 Fri 14:29

Emacs (<http://www.gnu.org/software/emacs/>) 28.2 (Org-mode (<http://orgmode.org>) 9.5.5)