DNSSEC

Tämä sivu sisältää:

Ohjeita DNSSEC-käyttöönottoon

Tähän dokumenttiin on kerätty ohjeita ja hyviksi havaittuja käytäntöjä DNSSEC:n käyttöönottoon, ylläpitokäytäntöihin ja valvontaan liittyen. Ohjeet ja suositukset perustuvat yleisesti saatavilla oleviin dokumentteihin (esim. RFC-dokumentit) sekä CSC/Funetin omiin kokemuksiin ja havaintoihin. On kuitenkin syytä muistaa, että kaikki suositukset eivät välttämättä ole sellaisinaan sopivia jokaiseen ympäristöön, vaan kullekin ympäristölle ominaiset tekniset reunaehdot sekä toimintamalleihin liittyvät käytännöt on aina otettava huomioon.

Tämän dokumentin tarkoituksena ei ole antaa yksityiskohtaista selvitystä nimipalvelusta tai DNSSEC:n toimintaperiaatteista. Dokumentissa annetaan erittäin lyhyt johdanto DNSSEC:iin, mutta tarkemmat yksityiskohdat löytyvät muualta, oleellisesti aiheeseen liittyvistä RFC-dokumenteista.

Mikä DNSSEC ja miksi?

DNSSECin avulla voidaan varmistaa, että DNS-vastaus on peräisin oikealta auktoritatiiviselta nimipalvelimelta ja että vastaus ei ole muuttunut siirtotiellä. Tällä tavalla voidaan suojautua tehokkaasti erityisesti niin sanotuilta Man-in-the-Middle -hyökkäyksiltä, joita vastaan perinteinen DNS-protokolla ei anna kovin vahvaa suojaa. Vastausten todentaminen perustuu julkisen avaimen kryptografiaan; DNS-tietueet allekirjoitetaan vyöhykkeen haltijan yksityisellä allekirjoitusavaimella ja todennetaan sitä vastaavalla julkisella avaimella. Allekirjoitusavainten yksityinen osa on tyypillisesti vain vyöhykkeen omistajan /haltijan/ylläpitäjän hallussa ja sen julkinen osa julkaistaan nimipalvelussa erityisen DNSKEY-tietueen muodossa.

DNSSEC-allekirjoitusten (RRSIG-tietueet) validointi perustuu luottamusketjuihin, jotka muodostetaan DNS-hierarkian mukaisesti. Käytännössä tämä tapahtuu siten, että allekirjoitetun vyöhykkeen julkinen avain allekirjoitetaan DNS-hierarkiassa ylemmällä tasolla olevan vyöhykkeen avaimella. Luottamusketjun huipulla on tyypillisesti internetin juurivyöhykkeen julkinen avain. Luottamusketjut muodostuvat julkisten avainten tiivisteistä muodostetuista DS-tietueista, jotka julkaistaan DNS-hierarkiassa ylempänä olevassa vyöhykkeessä ja allekirjoitetaan kyseisen delegoivan vyöhykkeen (parent-zone) avaimella.

DNS on todistetusti erittäin toimiva ja ennen kaikkea skaalautuva tietokanta ja DNSSEC-allekirjoittaminen mahdollistaa perinteisen nimipalvelutiedon lisäksi myös kriittisemmän tiedon julkaisemisen DNS:ssa. Esimerkiksi SSH-avainten sormenjälkien julkaiseminen DNS:ssa on mielekästä, jos sormenjälkitietueet (SSHFP-tietueet) voidaan allekirjoittaa ja validoida DNSSEC:n avulla.

Validoinnin käyttöönotto

DNSSEC-validoinnin avulla resolvoiva DNS-palvelin voi varmistaa vastaanottamiensa DNS-tietueiden alkuperän ja eheyden tietuiden allekirjoitusten avulla. Tässä kappaleessa kerrotaan, miten DNSSEC-validointi otetaan käyttöön resolvoivalla nimipalvelimella ja mitä asioita pitää erityisesti huomioida.

Nimipalvelinohjelmisto

Jotta nimipalvelin pystyy tekemään DNSSEC-validointia, siinä pitää olla asennettuna DNSSEC-validointia tukeva nimipalvelinohjelmisto. Ainakin Bindin ja Unboundin tiedetään tukevan DNSSEC-validointia, mikäli käytössä on riittävän uusi versio ohjelmistosta (Bindin osalta suositus 9.7.5. tai uudempi, Unbound 1.4.16 tai uudempi). Yleisesti ottaen on suositeltavaa käyttää laajasti käytössä olevia nimipalvelinohjelmistoja, sillä niihin on yleensä saatavilla tukea esimerkiksi muulta käyttäjäyhteisöltä. Koska DNSSEC on tuotantokäytössä edelleen verrattain uusi tekniikka, on odotettavaa, että nimipalvelinohjelmistoista löytyy DNSSEC-spesifisiä ongelmia ja mahdollisesti haavoittuvuuksiakin. Tämän vuoksi on ensiarvoisen tärkeää seurata käytössä olevan nimipalvelinohjelmiston bugi- ja haavoittuvuustiedotteita.

DNSSEC-allekirjoitusten voimassaololla on aina yksikäsitteinen alkamis- ja päättymisaika, mistä johtuen on äärimmäisen tärkeää, että validoivan resolverin sisäinen kello on oikeassa ajassa. Mikäli resolverin kellonaika on vääristynyt, se voi tulkita validit allekirjoitukset vanhentuneiksi tai toisinpäin. Tämän vuoksi on suositeltavaa synkronoida validoiva resolveri johonkin laadukkaaseen aikalähteeseen. Resolverin kellon synkronointiin voidaan käyttää esimerkiksi NTP- ja/tai PTP-protokollaa. NTP-palvelimet kannattaa konfiguroida nimipalvelimiin NTP-palvelinten DNS-nimien sijaan IP-osoitteita käyttäen, sillä tällä tavalla vältetään ristikkäiset riippuvuussuhteet; NTP-palvelinten DNS-nimet eivät resolvoidu, jos ne on allekirjoitettu ja resolverin kello on väärässä.

Koska virtuaalikoneiden ajan tahdistus on joskus ongelmallista, kannattaa validoiva resolveripalvelin mahdollisuuksien mukaan toteuttaa fyysistä dedikoitua palvelinlaitteistoa käyttäen. Koska nimipalvelu on verkon toiminnan kannalta kriittinen infrastruktuuripalvelu, ei sitä ole suositeltavaa toteuttaa virtualisointijärjestelmistä riippuvaisena virtuaalipalvelimena.

Validoiva resolveri ilmaisee auktoritatiiviselle nimipalvelimelle halustaan vastaanottaa DNSSEC-tietueita erillisen EDNS0-tietueen ja erityisesti siinä olevan DNSSEC OK (DO) -bitin avulla. EDNS0-tietueen avulla resolveri voi myös signaloida auktoritatiiviselle nimipalvelimelle pystyvänsä vastaanottamaan alkuperäisessä DNS-standardissa määriteltyä 512 tavua suurempia UDP-vastauksia (DNSSEC kasvattaa vastausten kokoa siten, etteivät ne yleensä mahdu 512 tavuun). UDP-vastauksen maksimikoko määritellään yleensä nimipalvelinohjelmiston konfiguraatiossa (Unboundissa "edns-buffer-size" ja Bindissa "edns-udp-size") ja suositeltava arvo sille on RFC6891:n suosituksen mukainen 4096 tavua, mikä on yleensä myös oletusarvo ohjelmistoissa. Jos resolverin palomuuri estää IP-fragmentit, kannattaa UDP-vastauksen maksimikoko konfiguroida Ethernet-verkoissa yleisesti käytössä olevaa 1500 tavun MTU:ta pienemmäksi (esim. 1480 tavua), jolloin fragmentoinnin tarpeen todennäköisyys pienenee. Lisää DNSSEC:n vaikutuksesta resolveripalvelimen palomuuraukseen myöhemmin tässä dokumentissa.

Luottamusankkuri

DNSSEC-validointi perustuu luottamusketjuihin, minkä vuoksi validoivalle nimipalvelimelle pitää konfiguroida ns. luottamusankkuri, joka toimii luottamusketjujen alkupisteenä. Luottamusankkuri on jonkun allekirjoitetun vyöhykkeen allekirjoitusavaimen julkinen osa (DNSKEY-tietue) tai siitä laskettu tiiviste (DS-tietue). Koska internetin juurivyöhyke on allekirjoitettu, on suositeltavaa konfiguroida juurivyöhykkeen julkinen avain tai sen tiiviste luottamusankkuriksi. Juuren luottamusankkurin voi hakea IANA:n verkkosivuilta ja **ennen käyttöönottoa sen aitous tulisi todentaa jollain DNS(SEC):n ulkopuolisella mekanismilla**, esimerkiksi PGP- luottamusverkostoja käyttäen. On myös hyvä varmistaa, että IANA:n sivuilta haettu luottamusankkuri vastaa nimipalvelusta löytyvää juurivyöhykkeen julkista avainta. Juuren DNSKEY-tietueen voi muuttaa DS-tiivisteeksi esimerkiksi LDNS-ohjelmistokirjastosta löytyvän ldns-key2ds -työkalun avulla (esim. "`dig . dnskey > root.dnskey && ldns-key2ds -2 -n root.dnskey`").

Tätä kirjoittaessa ei ole tiedossa, milloin juurivyöhykkeen julkinen avain tulee vaihtumaan, mutta ennemmin tai myöhemmin avain uusitaan. Tällöin myös validoivien resolvereiden luottamusankkurit pitää päivittää. RFC 5011:ssa on kuvattu mekanismi, jonka avulla validoiva resolveri voi päivittää luottamusankkurinsa automaattisesti. Koska vielä ei ole tiedossa, milloin ja erityisesti millä tavalla juurivyöhykkeen avain tullaan aikoinaan uusimaan ja miten edellä mainittu RFC5011-mekanismi käytännössä toimii, tulee juurivyöhykkeen avaimen vaihtumista valvoa myös muilla keinoin. On suositeltavaa rakentaa jonkinlainen valvontatyökalu, joka ilmoittaa validoivan resolverin ylläpitäjille, kun juurivyöhykkeen avain vaihtuu. Tällaisen työkalun voi toteuttaa yksinkertaisimmillaan cron-palvelussa ajettavana shell-skriptinä.

Palomuurit

DNSSEC-tietueet kasvattavat merkittävästi DNS-vastausten kokoa, minkä vuoksi ne eivät välttämättä mahdu enää alkuperäisessä DNS-standardissa määriteltyyn 512 tavuun. Nimipalvelimet signaloivat toisilleen kyvystään vastaanottaa 512 tavua suurempia UDP-vastausviestejä erityisen EDNS0-laajennuksen avulla. Jos resolveripalvelimen EDNS0-laajennuksen avulla signaloima UDP-vastauksen enimmäiskoko on suurempi kuin siirtotien MTU (Maximum Transmission Unit), auktoritatiivinen nimipalvelin joutuu lähettämään vastauksen fragmentoituna. **Tämän vuoksi on tärkeää, että resolverin palomuuri on konfiguroitu sallimaan fragmentit.**

Jos vastausviesti on niin suuri, ettei se mahdu EDNS0:n avulla signaloituun UDP-vastausviestin enimmäiskokoon, auktoritatiivinen nimipalvelin palauttaa lyhennetyn vastauksen UDP:lla ja merkitsee vastaukseen ns. Truncated-flagin. Truncated-flag saa aikaan sen, että resolveri tekee kysymyksen uudestaan TCP:tä käyttäen. DNSSEC:n myötä tällaiset tilanteet ovat entistä yleisempiä, **joten on tärkeää että resolverin palomuurissa sallitaan DNS-kyselyt myös TCP:llä**.

Jos resolveripalvelimen edessä on palomuuri, on myös tärkeää varmistaa, ettei se suodata Path MTU Discoveryn (PMTUD) tarvitsemia ICMP-virheilmoituksia. Erityisen tärkeää tämä on IPv6:n osalta, sillä IPv6-reitittimet eivät tee pakettien fragmentointia vaan hylkäävät liian suuret paketit ja lähettävät lähettävälle koneelle ICMP-virheilmoituksen.

Valvonta

DNSSEC muuttaa nimipalvelun toimintaa merkittävällä tavalla ja voi tuoda mukanaan myös uudenlaisia ongelmia, minkä vuoksi on ensiarvoisen tärkeää valvoa DNSSEC:n toimintaa sekä sille keskeisiä osakokonaisuuksia. DNSSEC-validoinnin osalta ainakin seuraavia asioita olisi syytä valvoa:

- Validointivirheiden määrä ja ei-validoituvien kohteiden selvittäminen tarvittaessa esimerkiksi nimipalvelinohjelmiston lokituksen tasoa lisäämällä
    - resolverin tilastojen saatavuus riippuu käytettävästä nimipalvelinohjelmistosta, mutta esimerkiksi Unboundissa tilastot (ml. validointivirheet) saa tulostettua "unbound-control stats" -komennolla
    - Huom! Kannattaa harkita tarkasti, haluaako validointivirheiden lokitusta pitää jatkuvasti päällä, sillä jatkuva ja kattava lokitus voi pahimmillaan kuormittaa resolveria huomattavasti ja avata täten uuden hyökkäysvektorin. Funetin validoivilla resolvereilla validointivirheiden jatkuva lokitus on pois päältä ja käännetään aina tarvittaessa tilapäisesti päälle, mikäli on tarpeen selvittää tarkemmin validointivirheiden syitä.
- SERVFAIL-vastausten määrä
- Resolvereiden ajan tahdistuksen tila (esim. NTP-synkronointi)

Erityisen tärkeää on pystyä havaitsemaan nopeasti, mikäli validointivirheiden määrässä tapahtuu äkillinen merkittävä muutos. Validointivirheiden valvontaa toteutettaessa on syytä suunnitella tarkoin valvonnalle määriteltävät raja-arvot, jotta jatkuvat validointivirheet eivät aiheuta turhia hälytyksiä ja pahimmassa tapauksessa peitä alleen laajempia ongelmia. Funetin resolveripalvelun valvonnassa raja-arvot on asetettu siten, että valvonta hälyttää, kun validointivirheiden määrä ylittää 1/sekunti (normaalitilanteessa Funetin resolvereilla tapahtuu validointivirheitä muutama per minuutti).

Omien vyöhykkeiden DNSSEC-allekirjoittaminen

Allekirjoitusjärjestelmän arkkitehtuuri

Allekirjoitusjärjestelmä toteutetaan tyypillisesti siten, että allekirjoituspalvelin sijoitetaan loogisesti nimipalvelutietojen hallintajärjestelmän ja julkisten nimipalvelinten väliin. Allekirjoitusjärjestelmä on tällöin eräänlainen "bump in the wire" -tyyppinen komponentti, jolla ei ole mitään ulospäin julkiseen internetiin näkyviä rajapintoja. Käytännössä allekirjoitusjärjestelmä saa allekirjoittamattomat zone-tiedostot hallintajärjestelmältä esimerkiksi AXFR/IXRF-rajapintaa ("zone transfer") tai jotain muuta (esim. SSH/SCP) rajapintaa käyttäen, allekirjoittaa zone-tiedostot ja siirtää sen jälkeen allekirjoitetut versiot zone-tiedostoista julkisille nimipalvelimille esimerkiksi jotain edellä mainittua rajapintaa käyttäen. Jos nimipalvelutietojen hallintajärjestelmä tukee DNSSEC-allekirjoittamista, allekirjoittaminen voidaan toteuttaa myös sen avulla, mikä yksinkertaistaa järjestelmän kokonaisarkkitehtuuria huomattavasti. Sen sijaan julkisella nimipalvelimella allekirjoitustoiminnallisuutta ei ole järkevä toteuttaa, sillä tällöin yksityisten allekirjoitusavainten suojaaminen on haastavampaa kuin silloin, kun allekirjoittaminen ja avainten säilytys tapahtuu järjestelmässä, jolla ei ole ulospäin auki olevia rajapintoja.

Ohjelmistovalinta on keskeisimpiä allekirjoitusjärjestelmän arkkitehtuuriin vaikuttavia tekijöitä. Ainakin avoimeen lähdekoodiin perustuva OpenDNSSEC on erittäin laajasti käytössä ja sitä käytetään jopa monen TLD-vyöhykkeen allekirjoittamiseen, joten sitä voidaan ainakin suositella. OpenDNSSEC:lla on myös erittäin aktiivinen kehittäjä- ja käyttäjäyhteistö, mikä tuo sille huomattavaa lisäarvoa erityisesti suljettuihin kaupallisiin ratkaisuihin verrattuna. Myös ainakin Bind-nimipalvelinohjelmiston uusimmissa versioissa on tuki ns. inline-signing toiminnallisuudelle, mutta sen käytännön toimivuudesta ei ole tätä kirjoittaessa tietoa. Periaatteessa tämä kuitenkin mahdollistaisi Bind-nimipalvelimen käyttämisen ns. hidden master palvelimena, jonka kautta allekirjoittamattomat zone-tiedostot siirtyvät allekirjoitettuina julkisille nimipalvelimille.

Allekirjoitusjärjestelmä on syytä kahdentaa vähintään sillä tasolla, että vikatilanteissa allekirjoitusavaimet ja toiminnallisuudet saadaan siirrettyä edes manuaalisesti kohtuullisessa ajassa varajärjestelmään. **Automaattisesti aktivoituvaa varajärjestelmää ei suositella käytettäväks**i, koska DNSSEC-allekirjoittamiseen liittyy muutamia asioita, jotka on hyvä tarkistaa ennen varajärjestelmän aktivoimista. Ennen varajärjestelmän aktivoimista on tärkeää varmistaa esimerkiksi se, että vara-allekirjoituspalvelin käyttää samoja avaimia kuin pääjärjestelmäkin ja että sillä on sama tilatieto avainten kierrätykseen yms. allekirjoitusparametreihin liittyen. Allekirjoituspalvelimen kahdentamisen tärkeyttä pohtiessa on syytä muistaa, että allekirjoitusjärjestelmän vikaantuminen ei itsessään aiheuta välitöntä katkoa nimipalvelun toiminnalle, mutta estää vyöhykkeen päivittämisen. Tämän seurauksena myöskään DNSSEC-allekirjoitukset eivät päivity, mikä tarkoittaa käytännössä sitä, että allekirjoitusjärjestelmä on saatava takaisin toimintakuntoon ennen kuin allekirjoitukset alkavat vanheta. Allekirjoitusjärjestelmän saatavuuden kriittisyyteen voi osaltaan vaikuttaa allekirjoitusparametrien järkevällä valinnalla, mistä lisää myöhemmin tässä dokumentissa.

Allekirjoitusjärjestelmän kahdentamisessa on muistettava varmistaa, että pää- ja vara-allekirjoituspalvelimella on aina samat avaimet käytettävissä. Jos avaimet on generoitu etukäteen, ne voidaan kopioida pääpalvelimelta varapalvelimelle järjestelmän käyttöönottovaiheessa, mutta jos avaimia generoidaan dynaamisesti tarpeen mukaan, on huolehdittava avainnipun synkronoinnista pääpalvelimelta varapalvelimelle. Avainten lisäksi myös vyöhykkeen allekirjoittamiseen liittyvä metatieto, joka sisältää mm. tiedon avainten uusimiseen liittyvistä ajoituksista, pitää olla synkronissa palvelinten välillä. Esimerkiksi OpenDNSSEC:ssa tämä tieto on ns. KASP-tietokannassa (Key and Signing Policy), joka on siis kahdennetuissa ympäristöissä pidettävä samansisältöisenä kummallakin palvelimella.

Allekirjoitusavaimet ja niiden hallinta

Koska DNSSEC perustuu julkisen avaimen kryptografiaan, on DNS-tietueiden allekirjoittamiseen käytettävän yksityisen avaimen luotettavuus koko järjestelmän perusta. Tämän vuoksi allekirjoitusjärjestelmää suunniteltaessa on syytä kiinnittää erityistä huomiota yksityisten avainten generointiin ja turvalliseen säilytykseen. Avainten generoinnissa eniten huomiota tulee kiinnittää siihen, että avaimet ovat mahdollisimman satunnaisia ja siten vaikeasti murrettavia. Avainten satunnaisuutta voidaan lisätä esimerkiksi käyttämällä erillistä ulkoista satunnaislukugeneraattoria, joita on saatavilla esimerkiksi USB-väylään kytkettävinä tikkuina. On myös mahdollista hyödyntää esimerkiksi uusimmista Intel-prosessoreista löytyvää RdRand-satunnaislukugeneraattoria, joka oleellisesti syöttää entropiaa käyttöjärjestelmän entropiavarastoon. Linux-palvelinten /dev/urandom -tiedostoa ei kannata käyttää entropialähteenä allekirjoitusavaimia generoitaessa, koska sen tuottamat satunnaisluvut eivät ole välttämättä riittävän satunnaisia.

Allekirjoitusavaimet, nimenomaan niiden yksityiset osat, tulee säilyttää mahdollisimman huolellisesti suojattuina siten, ettei kukaan ulkopuolinen pääse niihin käsiksi. Mikäli mahdollista, avaimet kannattaa säilyttää aina salattuina esimerkiksi erillisessä salatussa tiedostojärjestelmässä. Tästä on se haittapuoli, että allekirjoituspalvelimen käynnistyessä ylläpitäjän on syötettävä avainten säilyttämiseen käytettävän tiedostojärjestelmän salasana, mutta toisaalta tällä tavalla voidaan estää avainten väärinkäyttö esimerkiksi kiintolevyt varastamalla. Avaimia voidaan myös säilyttää erillisessä HSM-laitteessa (Hardware Security Module), jotka ovat kuitenkin yleensä melko kalliita ja vaativat jonkin verran erityisosaamista. HSM-laitteiden käyttäminen on perusteltua ylimmän tason vyöhykkeiden kuten maakohtaisten ccTLD-vyöhykkeiden allekirjoittamiseen ja avainten säilyttämiseen, mutta alemman tason vyöhykkeiden allekirjoittamiseen HSM-laitteen käyttäminen ei ole lainkaan välttämätöntä, mikäli allekirjoitusavainten satunnaisuus ja turvallinen säilytys varmistetaan muilla tavoin. Jos erillisen HSM-laitteen käyttö kiinnostaa, niin esimerkiksi OpenDNSSEC:n sivuilla on listattuna OpenDNSSEC-yhteensopivia HSM-laitteita. Sivuilta löytyy myös muuta lisätietoa HSM-laitteen valintaan liittyen.

Allekirjoitusavaimet on myös tärkeä muistaa varmuuskopioida, mieluiten siten että varmuuskopiot eivät ole selkokielisessä muodossa vaan esimerkiksi GnuPG-salattuina. On myös järkevä estää varmuuskopioimattomien avainten käyttö kokonaan, mikäli käytettävä allekirjoitusohjelmisto sellaista tukee. Esimerkiksi OpenDNSSEC voidaan konfiguroida siten, että se ei ota avaimia aktiiviseen käyttöön ennen kuin ne on varmuuskopioitu. Jos avaimia generoidaan dynaamisesti tarpeen mukaan, kannattaa varmuuskopiointi pyrkiä automatisoimaan siten, että aina kun allekirjoitusohjelmisto generoi uusia avaimia, avainnippu varmuuskopioidaan. Toinen vaihtoehto on generoida etukäteen allekirjoitusjärjestelmän käyttöönottovaiheessa suuri määrä avaimia ja ottaa niistä varmuuskopio.

Allekirjoitusavainten algoritmiksi RSA-algoritmi lienee tätä kirjoittaessa (lokakuu 2013) varmin valinta, koska sitä voidaan edelleen pitää turvallisena ja se on hyvin laajasti tuettuna. RSA-avainten pituudeksi voidaan suositella esimerkiksi 2048 bittiä. Tätä vahvemmille RSA-avaimille ei välttämättä ole perusteita, sillä internetin juurivyöhyke on allekirjoitettu 2048-bittisellä avaimella. Mikäli vyöhykkeen allekirjoittaminen on hyvin hidasta 2048-bittisellä RSA-avaimella, voidaan Zone Signing Key -avaimena käyttää myös 1024-bittistä RSA-avainta, erityisesti jos ZSK-avain uusitaan Key Signing Key -avainta tiheämmin. Avainten algoritmeista voidaan todeta vielä sen verran, että allekirjoitusjärjestelmän komponentit (ohjelmistot, mahdolliset lisälaitteet yms.) kannattaa valita siten, että ne tukevat myös elliptisen käyrän algoritmiin perustuvia avaimia eli ECDSA-avaimia, jotka tulevat todennäköisesti yleistymään tulevaisuudessa.

Tarkalleen ottaen DNS-tietueita ei allekirjoiteta sellaisenaan vaan niistä lasketaan ensin kryptografinen tiiviste, joka sitten allekirjoitetaan. Tiivisteen laskemiseen käytettävä algoritmi vaikuttaa osaltaan siihen, kuinka helppoa tai vaikeaa allekirjoituksen laskennallinen murtaminen on. Onkin suositeltavaa käyttää mahdollisimman vahvaa tiivistealgoritmia, vähintään SHA-256:a. MD5-tiivistealgoritmia ei missään nimessä suositella käytettävän tiedossa olevien heikkouksien vuoksi.

Avainten uusiminen säännöllisin väliajoin pienentää niiden laskennallisen murtamisen todennäköisyyttä. Avainten uusimisen helpottamiseksi suositellaan käytettäväksi kahta eri avainparia: Key Signing Key -avaimella (KSK) allekirjoitetaan vain vyöhykkeen avaimet, kun taas Zone Signing Key:tä (ZSK) käytetään vyöhykkeen varsinaisten DNS-tietueiden allekirjoittamiseen. KSK-avaimen uusiminen edellyttää interaktiota delegoivan vyöhykkeen (parent-zone) kanssa, kun taas ZSK-avaimen voi uusia ilman toimenpiteitä delegoivan vyöhykkeen suuntaan. ZSK-avaimen uusiminen kannattaakin ehdottomasti jättää allekirjoitusohjelmiston tehtäväksi. Päivitysintervalliksi riittää ZSK-avaimen osalta esimerkiksi 4 kertaa vuodessa. KSK-avaimen uusimisessa on kahta koulukuntaa: joidenkin mielestä KSK-avainta ei kannata uusia kuin allekirjoitusjärjestelmää/algoritmia tms. uusittaessa, kun taas osa perustelee KSK-avaimen uusimista sillä, että vain tällä tavalla uusimisprosessi pysyy mielessä ja tulee säännöllisesti harjoiteltua. Jos KSK-avainta halutaan uusia säännöllisesti, 1-2 vuoden intervallia on yleensä pidetty sopivana. Jos KSK-avainta ei haluta uusia säännöllisesti, voi olla järkevä käyttää vahvempaa, esimerkiksi 4096-bittistä avainta.

Parametrien valinta

RFC6781:ssa ja RFC-draftissa draft-ietf-dnsop-dnssec-key-timing on annettu hyviä suosituksia DNSSEC:iin liittyvien allekirjoitus- ja aikaparametrien valintaan. DNSSEC-allekirjoitusjärjestelmän pystyttämistä suunnittelevan kannattaa perehtyä edellä mainittuihin dokumentteihin, mutta tässä kappaleessa korostetaan muutamia keskeisimpiä hyväksi havaittuja ohjenuoria ja huomioitavia asioita.

DNSSEC-parametrien valinnassa keskeisin asia on varmistaa, että erilaiset ajoituksiin liittyvät parametrit ovat sellaisia, että mahdollisissa vikatilanteissa ylläpitäjillä on riittävästi aikaa havaita ja korjata ongelma ennen kuin vyöhykkeen allekirjoitukset alkavat vanheta. Parametrien valinta vaikuttaa suoraan esimerkiksi siihen, kuinka kauan on aikaa havaita ja korjata allekirjoitusjärjestelmässä tai muussa kohtaa prosessiketjua ilmennyt vika. Jos esimerkiksi allekirjoitusten voimassaoloaika on oletuksena vain kaksi vuorokautta, allekirjoitusjärjestelmän vikaantuessa kaikki allekirjoitukset vanhenevat kahden vuorokauden sisällä. Allekirjoitusten voimassaoloajan lisäksi myös allekirjoitusten uusimisväli vaikuttaa allekirjoitusten vanhenemiseen; jos allekirjoitukset uusitaan aina esimerkiksi 10 vrk ennen niiden vanhenemista, niin allekirjoittamisen jälkeen vyöhykkeessä ei ole koskaan allekirjoituksia, jotka vanhenisivat alle 10 vuorokauden kuluttua.

Ajoituksiin liittyvissä parametreissa tulee huomioida se, kuinka nopeasti ylläpitäjät ovat saatavilla korjaamaan ongelmia, ottaen huomioon myös viikonloput ja lomakaudet. Melko turvallisina asetuksina voidaan pitää esimerkiksi sellaisia, että allekirjoitusten voimassaoloaika on aina 14 vuorokautta ja ne uusitaan viimeistään silloin, kun niiden voimassaoloaikaa on jäljellä 10 vuorokautta. Tämä antaa käytännössä 10 vuorokautta aikaa havaita ja korjata vyöhykkeen ja erityisesti sen allekirjoitusten päivittymisen estävän vian. Toki on myös muistettava allekirjoittamissyklin tiheyden vaikutus: jos allekirjoitusprosessi käynnistetään esimerkiksi vain kerran vuorokaudessa, voi vyöhykkeessä olla juuri ennen allekirjoittamista 9 vuorokauden päästä vanhenevia allekirjoituksia, jos käytetään edellä mainittua 10 vuorokauden päivitysraja-arvoa allekirjoituksille. Yleensä onkin järkevä allekirjoituttaa vyöhyke huomattavasti tätä tiheämmin, esimerkiksi kerran tunnissa, sillä silloin on myös helpompi valvoa koko allekirjoitusprosessin toimivuuttakin. Säännöllisten, määritellyn aikajakson välein tapahtuvien allekirjoittamisten lisäksi vyöhyke pitää allekirjoittaa luonnollisesti myös aina silloin, kun sen tietoja muutetaan. Säännöllinen allekirjoittaminen tehdään sen vuoksi, että jos vyöhykkeen tietoja ei muuteta pitkään aikaan, allekirjoittamisprosessi (sis. uudelleenallekirjoittamista vaativien tietueiden läpikäynti, avainten uusimistarpeen evaluointi yms.) käynnistyy silti tasaisin väliajoin.

DNSSEC:lla on vaikutusta myös vyöhykkeen SOA-tietueen arvoihin, sillä SOA-tietue mm. kertoo slave-palvelimille, kuinka useasti niiden pitää tarkistaa master-palvelimelta, onko vyöhyke päivittynyt ja kuinka pitkän ajan kuluttua niiden pitäisi tulkita vöyhyke expiroituneeksi, jos master-palvelin ei ole tavoitettavissa. DNSSEC vaikuttaa näiden arvojen valintaan siten, että vyöhykkeen pitäisi antaa mieluummin expiroitua, kuin että nimipalvelin palauttaisi vanhentuneita allekirjoituksia. Jos vyöhyke expiroituu, auktoritatiivinen nimipalvelin palauttaa resolverille SERVFAIL-vastauksen, jolloin resolverin pitäisi osata kysyä joltain toiselta auktoritatiiviselta palvelimelta. Jos taas vyöhyke ei ehdi expiroitumaan ennen allekirjoitusten vanhenemista, auktoritatiivinen palvelin palauttaa resolverille vanhan allekirjoituksen, jonka validoiva resolveri tulkitsee luonnollisesti vääräksi ja palauttaa alkuperäiselle kyselijälle SERVFAIL-vastauksen ja alkuperäisen kyselijän tyypistä riippuen (forwardoiva nimipalvelin vs. käyttäjän käyttöjärjestelmän resolverkirjasto) tämä saattaa luovuttaa siihen. Tämän vuoksi on suositeltavaa asettaa vyöhykkeen SOA expiry-ajastin kertaluokkaa pienemmäksi kuin mikä allekirjoitusten voimassaoloaika on.

DNSSEC mahdollistaa myös negatiivisten vastausten todentamisen, eli käytännössä tiedon siitä että jotain kysyttyä nimeä tai tietuetta ei ole olemassa. Tämä tapahtuu erityisten NSEC-tietueiden avulla. NSEC-tietueilla on kuitenkin sellainen negatiivinen sivuvaikutus, että ne mahdollistavat vyöhykkeen kaikkien tietojen listaamisen triviaalisti. Vaikka nimipalvelussa oleva tieto on lähtökohtaisesti julkista, voi vyöhykkeen kaikkien tietojen listaamisen mahdollisuus olla joissain tilanteissa haitallista. Tämän vuoksi kannattaa käyttää NSEC:n sijaan NSEC3-tietueita. Poikkeuksena sellaiset vyöhykkeet, joissa on vain muutama julkinen tietue (esim. pelkkä www-tietue tai CNAME); näiden osalta NSEC3:n käyttäminen ei tuo juurikaan lisäarvoa NSEC:iin verrattuna. NSEC3:ssa on myös ns. OptOut-ominaisuus, mutta se on merkityksellinen lähinnä vain suurissa, pääasiassa delegointeja sisältävissä vyöhykkeissä. Tyypillisen toisen tason vyöhykkeen (esim. funet.fi) allekirjoittamisessa OptOut:n käyttäminen ei ole tarpeen.

DS-tietueiden päivitys

Jotta allekirjoitettu vyöhyke voidaan DNSSEC-validoida, vyöhykkeen KSK-avainta vastaava DS-tietue täytyy julkaista delegoivassa vyöhykkeessä ja allekirjoittaa delegoivan vyöhykkeen (ZSK-)avaimella. FI-päätteisten domainien osalta DS-tietue pitää siis julkaista fi.-juuressa, mikä tapahtuu Viestintäviraston verkkotunnusjärjestelmän kautta. Palveluntarjoajille on saatavilla Web Service -rajapinta tietojen päivittämisen automatisointia varten, mutta muiden on päivitettävä DS-tietueet manuaalisesti web-käyttöliittymän kautta. Koska tämä pitää tehdä kuitenkin vain käyttöönottovaiheessa ja KSK-avaimen uusimisen yhteydessä, ei tämän pitäisi aiheuttaa kohtuuttomasti lisävaivaa.

Käänteisvyöhykkeiden (reverse) osalta delegoiva vyöhyke on tyypillisesti Funetin ylläpidossa oleva vyöhyke, poislukien PA- tai legacy-osoiteavaruudet, joita vastaavan käänteisvyöhykkeen delegoiva vyöhyke on yleensä suoraan jonkun alueellisen osoiterekisterin (RIR, esim. RIPE tai Arin) ylläpidossa oleva käänteisvyöhyke. Funetin käänteisvyöhykkeitä ei ole vielä (lokakuussa 2013) allekirjoitettu, minkä vuoksi myös mahdollinen rajapinta DS-tietueiden päivittämistä varten puuttuu. Jos delegoiva vyöhyke on suoraan esim. jonkun RIR:n ylläpidossa, DS-tietueiden päivittäminen tapahtuu kyseisen RIR:n tarjoamien rajapintojen kautta. Esimerkiksi RIPE:n osalta DS-tietueita voi päivittää Webupdates-rajapinnan tai automaattisen sähköpostirajapinnan kautta.

Valvonta

Jos valvonta on tärkeää DNSSEC-validoinnin osalta, se on jopa vielä tärkeämpää allekirjoitettujen vyöhykkeiden saatavuuden varmistamiseksi. DNSSEC-allekirjoitetun vyöhykkeen osalta tulisi valvoa ainakin seuraavia asioita:

- Vyöhyke DNSSEC-validoituu. Jos valvontajärjestelmän käyttämä resolveri tukee DNSSEC-validointia, tämä tarkistus voidaan toteuttaa yksinkertaisesti laittamalla DNS-kyselyyn DO-bitti päälle ja tarkistamalla vastauksesta, että siinä on AD-bitti (Authenticated Data) päällä. Tarkistustyökalun tarkempi tekninen toteutus riippuu toki käytettävästä valvontaohjelmistosta, mutta esimerkiksi Nagiokseen on saatavilla erilaisia DNSSEC-laajennuksia.

- Allekirjoitusten voimassaolon jäljellä oleva aika. Normaalitilassa jäljellä olevan ajan pitäisi piirtää sahalaitakuviota; se pienenee aina kohti määriteltyä allekirjoituksen päivittämisen raja-arvoa (esim. 10vrk) ja hyppää siitä aina allekirjoituksille määriteltyyn voimassaoloaikaan (esim. 14vrk), kun allekirjoitus uusitaan.
- Vyöhykkeen viimeisin päivitys: allekirjoittamaton vyöhyke voi olla päivittymättä vaikka vuosia, mutta allekirjoitetun vyöhykkeen pitää päivittyä säännöllisin väliajoin, jotta allekirjoitukset eivät vanhene. Päivitysintervalli määräytyy annettujen parametrien mukaan, mutta kun ne on tiedossa, tulee valvonnan keinoin varmistaa että vyöhykkeen viimeisimmästä päivittymisestä ei ole kulunut pidempään kuin kuuluisi. Valvontalogiikka voidaan toteuttaa esim. seuraavilla tavoilla:
    - Jos SOA-sarjanumerona käytetään unix-aikaleimaa, saadaan vyöhykkseen viimeisimmän päivityksen ajankohta suoraan SOA-sarjanumerosta
    - Jos käytetään YYYYMMDDXX-muotoista SOA-sarjanumeroa, siitä ei suoraan nähdä viimeisimmän päivityksen ajankohtaa. Koska SOA-tietue ja sitä myöten myös sen allekirjoitus kuitenkin päivittyy jokaisen vyöhykkeen muutoksen yhteydessä, voidaan viimeisimmän päivityksen ajankohta lukea SOA-tietueen allekirjoituksen voimassaolon alkuhetkestä (inception timestamp). Tässä pitää kuitenkin huomioida allekirjoitusjärjestelmän mahdollisesti käyttämä turvaväli, sillä esimerkiksi OpenDNSSEC määrittelee allekirjoituksen voimassaolon alkuhetkeksi [T_{nyt} - 3600s]
- Vyöhykesiirtojen (zone transfer) toimivuus allekirjoitusjärjestelmän ja julkisten nimipalvelinten (tai julkisen master-palvelimen ja slave-palvelinten) välillä. Oleellista on huomata, mikäli vyöhyke ei päivity tarpeeksi nopeasti joillekin nimipalvelimille. Jos vyöhykkeen viimeisintä päivitystä valvotaan kuten edellisessä kohdassa on mainittu, tämän osalta riittää valvoa että kaikilla auktoritatiivisilla nimipalvelimilla on sama versio vyöhykkeestä.

On myös tärkeää tehdä muutamia tarkistuksia vyöhykkeen allekirjoittamisen jälkeen ennen sen päivittämistä julkisille nimipalvelimille. Erityisesti kannattaa tarkistaa, että allekirjoitetun vyöhykkeen tietueet validoituvat vyöhykkeen omia avaimia vasten (self-validation) ja että vyöhykkeen avaimet on allekirjoitettu KSK-avaimella, jota vastaava DS-tietue löytyy delegoivasta vyöhykkeestä. Esimerkiksi OpenDNSSEC:ssa tällaiset tarkistukset pystyy toteuttamaan melko suoraviivaisesti määrittelemällä konfiguraatioon NotifyCommand-optio, jonka avulla voidaan liipaista haluttu tarkistusskripti. Edellä mainittujen tarkistusten lisäksi tai osittain niiden toteuttamiseen voidaan käyttää myös valmiita saatavilla olevia työkaluja, kuten ValiDNS:aa.

Aiheeseen liittyviä RFC-dokumentteja

RFC 4033: DNS Security Introduction and Requirements

RFC 4034: Resource Records for the DNS Security Extensions

RFC 4035: Protocol Modifications for the DNS Security Extensions

RFC 6781, DNSSEC Operational Practices, Version 2

RFC 6891, Extension Mechanisms for DNS (EDNS(0))

Linkkejä DNSSEC testaustyökaluihin

https://www.dns-oarc.net/oarc/services/replysizetest

http://http://dnstest.ficora.fi

http://dnssec-debugger.verisignlabs.com/