ELEKTROTEHNIŠKI VESTNIK 79(1-2): 47-54, 2012
EXISTING SEPARATE ENGLISH EDITION

Upravljanje ključev v DNSSEC

Dušan Kozic[1], Benjamin Zwittnig[1], Janez Sterle[2], Andrej Kos[2]

[1] *Arnes, Jamova 39, 1000 Ljubljana, Slovenija*
[2] *Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, 1000 Ljubljana, Slovenija*
E-pošta: dusan.kozic@arnes.si

Povzetek. Varnostna razširitev protokola DNS, DNSSEC, je bila standardizirana leta 2005, od dopolnitve leta 2008 pa je primerna za širšo uporabo. Implementacija DNSSEC je kompleksno opravilo, ki zahteva spremembe na programski in strojni opremi v celotni hierarhiji DNS, zato se je DNSSEC začel uveljavljati šele v zadnjem času. V članku smo predstavili in med seboj primerjali trenutne možnosti implementacije DNSSEC, ki jih imajo na voljo ponudniki storitev DNS. Ugotovili smo, da so trenutno dostopna orodja dovolj zmogljiva, da jih lahko začnemo vpeljevati v širšo uporabo.

Ključne besede: DNSSEC, kriptografija z javnim ključem, digitalno preverjanje, podpisana DNS-zona, HSM

DNSSEC Key Management

The DNS security extensions, DNSSEC, were standardized in 2005. Since the 2008 update, they have become available for general use. The implementation of the DNSSEC is a complex task, demanding software and hardware modifications throughout the entire DNS hierarchy. That is the reason why DNSSEC has only recently received more attention. The paper presents and compares current possibilities for DNSSEC implementation, which are available to DNS service providers. The authors believe that the currently accessible tools are powerful enough for widespread use.

1 UVOD

Informacijski in telekomunikacijski sistemi ter storitve so še vedno ena najhitreje rastočih gospodarskih panog. Glavni razlog za to je hiter razvoj novih tehnologij in poslovnih modelov, ki vključujejo jedrna, agregacijska in dostopovna omrežja, uporabniško opremo, strežnike ter širok nabor različnih storitev. Pod terminom storitve v zadnjem času razumemo predvsem storitve za končne uporabnike [9],[17],[19]-[22], izjemno pomembne pa so tudi sistemske in varnostne storitve [8],[18]. Ena ključnih sistemskih storitev paketnih omrežij IP (angl. Internet Protocol) je tudi sistem domenskih imen (angl. DNS – Domain Name System), ki v osnovi ni podpiral mehanizmov za zagotavljanje varnostnih storitev.

Varnostni mehanizmi v DNS, ki jih uporabljamo danes, prvotno niso bili mišljeni kot funkcije za zagotavljanje varnosti. Izbire naključnih izvornih vrat se poslužujejo vse aplikacije, identifikator transakcije pa je bil namenjen povezovanju zahtevkov DNS z odgovori DNS. Tako je sistem DNS, ki je eden temeljnih sistemov za delovanje Interneta, ranljiv za vrsto napadov: od prisluškovanja komunikaciji do aktivnega spreminjanja vsebine s strani vrinjenega napadalca in zastrupljanja medpomnilnika DNS. [6]

Za zaščito DNS so leta 1995 začeli razvijati varnostne razširitve protokola DNS (angl. DNSSEC – Domain Name System Security Extensions). Zadnja specifikacija je bila sprejeta leta 2005 (RFC 4033-4035 [11][12][13]). Osnovni standard, ki naj bi v sistem DNS vnesel varnostne razširitve, je vseboval eno večjih varnostnih pomanjkljivosti, tj. možnost, da se napadalec lahko dokoplje do vsebine celotne zone. Ko je bil leta 2008 (RFC 5155 [15]) ustrezno popravljen, je postal zrel za produkcijsko vpeljavo. Tako je bila julija 2010 podpisana korenska zona, v zadnjem času pa je bila podpisana večina generičnih vrhnjih domen. Podpisujejo se tudi vrhnje domene, ki pripadajo različnim državam. V tem trenutku sicer še ni podpisanih veliko domen, ki so v lasti organizacij in posameznikov, hkrati pa je tudi težko napovedati, v kolikšnem obsegu bo DNSSEC na njih zaživel.

DNSSEC že tako zahtevnemu sistemu DNS dodaja še dodaten nivo kompleksnosti, ki ga pomeni uporaba kriptografije z javnim ključem (PKI – angl. Public Key Criptography). Vsi odgovori DNS so tako digitalno podpisani, medtem ko sporočila ob prenosu niso šifrirana. DNSSEC nam torej zagotavlja avtentikacijo in integriteto, zaupnosti komunikacije pa ne. Ob vpeljavi DNSSEC postaja delo z DNS bistveno bolj zapleteno in kompleksno, prav tako je možnost napak bistveno večja. Čeprav še ni širšega povpraševanja po storitvah DNSSEC, pa posamezniki že posegajo po njem. Zato se ponudniki internetnih storitev ne bodo mogli izogniti implementaciji in uporabi DNSSEC v svojih okoljih.

V članku smo se lotili najtežjih opravil DNSSEC, to so opravila, povezana z vzdrževanjem zon in ključev, s katerimi so le-te podpisane. V uvodu je opisan problem

Prejet 23. april, 2012
Odobren 7. maj, 2012

vzdrževanja ključev in zon. V drugem poglavju sledi osnovni opis izbranih orodij za to delo. Naredili smo primerjavo dveh plačljivih in enega odprtokodnega orodja. V tretjem poglavju se nahajajo rezultati tehnične primerjave orodij, ki smo jo izvajali ob podpisovanju ene velike in 400 srednje velikih zon, s čimer smo simulirali okolje manjšega ponudnika internetnih storitev. Na koncu je izvedena še primerjava orodij glede prijaznosti do uporabnika in nadzora nad delovanjem sistema.

2 VZDRŽEVANJE ŠIFRIRNIH KLJUČEV IN ZON

Pri upravljanju manjših zon je postavitev sistema DNS tipično pomenila enkraten poseg v sistem. Administrator sistema je moral posredovati samo ob spremembah in lahko se je zgodilo, da se mu z določenimi zonami ni bilo treba ukvarjati več let.

DNSSEC mora zagotavljati varnostne storitve z ustrezno varnostno politiko, zato datoteka zone ni več statična:

- podpis zone ima omejen rok trajanja, kar pomeni, da je treba zono znova podpisati, preden ključi potečejo;
- ključa KSK (angl. Key Signing Key) in ZSK (angl. Zone Signing Key) je treba periodično menjavati[1].

Če ne bi redno menjavali šifrirnih ključev, bi jih lahko na dolgi rok razkrili. Sodobna kriptografija namreč temelji na predpostavki, da za razkrivanje šifer po metodi preizkušanja vseh kombinacij potrebujemo le dovolj časa in veliko procesorsko moč. Sčasoma se torej poveča verjetnost, da bodo šifrirni ključi razkriti.

2.1 Menjava šifrirnih ključev

Ponovno podpisovanje zone je dokaj preprosto, saj od administratorja zahteva zgolj, da znova podpiše zono z veljavnimi ključi, kar v obstoječih rešitvah DNS praviloma lahko naredi z enim ukazom oz. enim klikom.

Menjava ključev pa je bolj zapleteno opravilo, saj je treba pred ponovnim podpisovanjem zone ključe na ustrezen način generirati, stare pa na koncu na varen način uničiti.

Pravzaprav pa to ni dovolj, saj sistem DNS temelji na medpomnjenju odgovorov za določen čas TTL. Ko rekurzivni strežnik dobi odgovor DNS, si ga zapomni za čas, kot ga določa čas TTL. To pomeni, da bo strežnik znova šel v poizvedovanje šele, ko bo čas TTL potekel. Če bi ob menjavi ključa v sistemu DNS zono podpisali zgolj z novim ključem, bi lahko prišlo do naslednjega scenarija:

1) Uporabnik na rekurzivnem strežniku poizveduje po imenu www.dnssec.si.
2) Rekurzivni strežnik tega imena nima v medpomnilniku, zato sprašuje naprej.
3) Dobi odgovor in digitalni podpis za ime www.dnssec.si.
4) Ključ DNSKEY za dnssec.si rekurzivni strežnik že ima v svojem medpomnilniku, saj so drugi uporabniki že prej spraševali po domeni, zato ne bo sprožil ponovnega povpraševanja po njem.

Podpis zapisa www.dnssec.si bo tako rekurzivni strežnik preverjal s starim ključem DNSKEY in varnostno preverjanje bo dalo negativen rezultat.

2.2 Menjava ključa ZSK

Obstajata dva načina menjave ključev: metoda po predhodni objavi (angl. pre-publish) in metoda dvojnega podpisovanja.

Z metodo predhodne objave nov ključ objavimo, preden z njim podpišemo zono. Postopek je naslednji:

- vsaj za dvakratni čas TTL[2], preden poteče zona, objavimo skupaj s starim ključem DNSKEY tudi novega,
- zono še vedno podpišemo zgolj s starim ključem,
- po izteku časa TTL zono podpišemo zgolj z novim ključem, vendar starega še vedno pustimo v zoni,
- po dvakratnem času TTL stari ključ umaknemo iz zone.

Z metodo dvojnega podpisovanja zono podpišemo hkrati z novim in starim ključem. Po času TTL iz zone umaknemo stari ključ in zono podpišemo zgolj z novim.

Za menjavo ključa ZSK je ne glede na dvakratni potreben čas TTL bolj praktična metoda predhodne objave. Datoteka podpisane zone pri tej metodi je namreč dvakrat manjša, kakor bi bila pri metodi dvojnega podpisovanja, sporočila DNS pa so prav tako dvakrat krajša. Metoda z dvojnim podpisovanjem je nujna, če s ključem zamenjamo tudi algoritem, s katerim podpisujemo sporočila DNS[1].

2.3 Menjava ključa KSK

Pri menjavi ključa KSK moramo biti pozorni, da ob menjavi strežniku starševske domene sporočimo zapis DS. Tako je treba upoštevati čas TTL zapisa DS, ki je običajno daljši od časa TTL naše zone[3]. Pri menjavi ključa KSK uporabljamo metodo dvojnega podpisovanja, saj bi bila uporaba metode enojnega podpisovanja časovno potratna, medtem ko pri velikosti odgovorov DNS ne bi veliko prihranili, ker je s ključem KSK podpisan zgolj ključ ZSK. Postopek menjave je naslednji:

- vsaj za čas TTL, preden poteče zona oz. zapis DS v starševski zoni, objavimo nov ključ KSK in njegov izvleček DS v starševski zoni,
- ključ ZSK podpišemo z novim in starim ključem,

[1] Ključ ZSK uporabimo za podpisovanje vseh zapisov v zoni, ključ KSK pa za podpisovanje ključa ZSK.

[2] Za vrednost časa TTL vzamemo najdaljši čas TTL v zoni.

[3] Vedno je treba upoštevati najdaljši čas TTL.

- po času TTL iz zone umaknemo stari ključ KSK in iz starševske zone prav tako izvleček DS tega ključa. [1]

3 METODOLOGIJA IN TESTNO OKOLJE

Da bi ponudniki storitev laže prešli na DNSSEC, smo se odločili primerjati različna orodja, ki so začela nastajati z razvojem DNSSEC. Med orodji smo se odločili primerjati njihove tehnične zmogljivosti in uporabniške izkušnje. Med tehničnimi zmogljivostmi so nas zanimali:

- zanesljivost delovanja,
- hitrost delovanja,
- možnost postavitve v visoko razpoložljivem načinu,
- skladnost z varnostnimi standardi FIPS (angl. Federal Information Processing Standard)[5],
- podprtost različnih algoritmov DNSSEC,
- prehod med algoritmi DNSSEC.

Med uporabniškimi zahtevami, pa so nas je zanimali:

- preprostost uporabe,
- količina znanja o DNSSEC, ki se pričakuje od uporabnika,
- podpora vmesnikom API,
- nadzor nad opravili DNSSEC,
- podpora proizvajalca.

Administratorju posamezne domene, ki uporablja DNSSEC in ima lastne strežnike DNS, delo z vzdrževanjem zone ne bo povzročalo preglavic. Lahko pa si zamislimo tudi slednji scenarij pri ponudnikih storitev DNS. Ponudniki storitev DNS bodo morali večjemu številu uporabnikov ponuditi kakovostno storitev DNSSEC, ki bo vključevala:

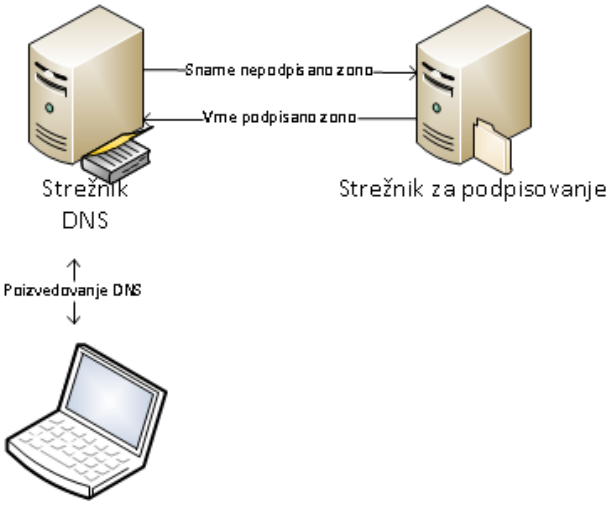
- samodejno ponovno podpisovanje zone,
- samodejno menjavo ključev ZSK in KSK,
- avtomatizirano sporočanje izvlečkov DS starševski zoni,
- možnost prehoda med različnimi algoritmi DNSSEC,
- možnost prehoda med različnimi orodji DNSSEC.

3.1 Postavitev okolja

Postavili smo testno okolje, ki je vsebovalo veliko zono s 100.000 delegacijami na različne strežnike DNS in 400 srednje velikih zon, ki so bile delegirane na istem strežniku DNS. Zona s sto tisoč delegacijami je bila velika 100 MB, posamična srednje velika zona pa je bila velika približno100 kB.

Primerjali smo odprtokodno orodje OpenDNSSEC verzije 1.3.0 [10], med plačljivimi orodji pa smo preizkusili rešitev Infoblox, verzije 5.1 [7], BlueCat Proteus, verzije 3.5 ter BlueCat Adonis, verzije 6.5 [4]. OpenDNSSEC je namensko orodje za podpisovanje DNSSEC, medtem ko sta BlueCat in Infoblox t. i. rešitvi IPAM (angl. Internet Protocol and Address Management), ki združujeta funkciji strežnika DHCP in strežnika DNS.

Logična shema testnega okolja (Slika 1.) je bila sestavljena iz dveh enot: strežnika za podpisovanje in strežnika DNS, ki je stregel podpisane zone.



Slika 1: Logična shema testnega okolja DNSSEC

Postavljeno okolje se je pri različnih orodjih nekoliko razlikovalo. Pri orodjih OpenDNSSEC in Infoblox sta se strežnik za podpisovanje in strežnik DNS poganjala na istem računalniku, na katerem je bil nameščen operacijski sistem Linux. Za uporabo orodja BlueCat pa sta bila potrebna dva strežnika: BlueCat Proteus je bil osrednji strežnik, namenjen nadzorovanju strežnika Adonis, ki je bil aktiven strežnik DNS. Zone, politika podpisovanja in ključi, pa so se nahajali na strežniku Proteus, ki je izvajal redistribucijo na strežnik Adonis.

Obe plačljivi orodji, Infoblox in BlueCat, temeljita na sistemu Linux in za strežnik DNS uporabljata najbolj razširjeni odprtokodni strežnik Bind [3]. Prav tako za podpisovanje zon uporabljata orodja tega strežnika, ki tudi skrbi za ponovno podpisovanje zon pred potekom veljavnosti. Za preverjanje veljavnosti podpisanih zon ne uporabljata nobenih dodatnih orodij. Zone preveri samo strežnik Bind, ki to stori, preden jih naloži. Za nadzor nad politiko DNSSEC, ki vključuje generiranje novih ključev ter menjavo ključev ZSK in KSK, pa orodji skrbita sami. Podatki (zone, ključi) so shranjeni v bazi podatkov.

Za primerjavo delovanja smo preizkusne verzije orodij postavili kot virtualne računalnike v okolju VMware ESX. Preizkusna licenca orodja Infoblox je imela omejeno procesorsko moč na 2000 MHz in pomnilnik na 2 GB RAM, zato smo tudi drugim orodjem nastavili to omejitev. Dejansko sta orodji BlueCat in Infoblox namenski strojni napravi, vendar proizvajalca omogočata njihovo testiranje na virtualnih računalnikih, kar smo storili tudi mi.

3.2 Visoka razpoložljivost

Zelo pomembno je, da imamo pri okvar katerega izmed strežnikov na redundantni lokaciji zagotovljeno čimprejšnje nadaljevanje vzdrževanja zon DNSSEC.

Orodje BlueCat podpira metodo XHA (angl. Crossover High Availability). V tem primeru sta med seboj povezani dve enaki napravi BlueCat, ki komunicirata prek dodeljenega omrežnega vmesnika. Če odpove katera od naprav, je prehod na sekundarno samodejen. Prav tako BlueCat podpira replikacijo baze podatkov na naprave BlueCat, ki so v pripravljenosti. Bazo je mogoče replicirati na več naprav, ki so lahko na različnih lokacijah.

Orodje Infoblox je zasnovano na t. i. gručah (angl. grid). Ena izmed naprav je gospodar (angl. Master), ki svoje podatke naprej replicira na druge člane gruče. Za gospodarja lahko podobno kot pri orodju BlueCat pripravimo visokorazpoložljivostni par. Dodatno lahko definiramo druge vloge, ki jih imajo druge naprave v gruči oz. kateri servisi naj tečejo na njih.

Rešitev OpenDNSSEC v osnovi ni zasnovana za delovanje v visokorazpoložljivostnem načinu, lahko pa takšno okolje postavimo sami. Nastavitev strežnika je lahko shranjena v bazi MySQL, ki podpira replikacijo na drug sistem. Za ključe, ki se nahajajo na HSM, lahko med strojnimi HSMi izberemo takšnega, ki deluje v visokorazpoložljivostnem načinu, pri SoftHSM pa lahko periodično redistribuiramo bazo ključev. V OpenDNSSEC lahko prav tako ključe za neko obdobje (npr. pet let) generiramo vnaprej in jih prekopiramo na sekundarni sistem. Tako imamo postavljen sekundarni sistem, ki je v pripravljenosti, da ob morebitnem izpadu primarnega sistema prevzame njegovo vlogo[4].

3.3 Varnost

V sodobni kriptografiji je varnost podatkov odvisna od dolžine, naključnosti in tajnosti ključev. Pri DNSSEC je tako zelo pomembno, kje so shranjeni naši zasebni ključi. Vdora v strežnik DNS nikoli ne moremo popolnoma izključiti, zato ni priporočeno, da bi bili ključi shranjeni na njem.

Za varno shranjevanje občutljivih podatkov obstaja standard FIPS 140-X [5]. FIPS določa zahteve, ki jih morajo izpolnjevati kriptografski moduli. Med drugim tudi, kje se lahko nahajajo šifrirni ključi, kako se generirajo in kako se na koncu uničijo. Naprave, ki zagotavljajo dovolj visoko stopnjo varnosti, morajo biti ustrezno certificirane. Orodji BlueCat in Infoblox nista certificirani po FIPS. Oba proizvajalca sicer trdita, da je zasnova orodja v skladu s standardom FIPS. Pri omenjenima orodjema je sporno, da ključi niso zgolj shranjeni v bazi podatkov, ampak tudi v datotekah strežnika Bind. Bind ima ključe na voljo zaradi dinamičnega DNS, pa tudi zato, ker sam skrbi za ponovno podpisovanje zon. V orodju BlueCat, kjer poteka podpisovanje zon na strežniku Adonis, to pomeni, da se hkrati z zonami periodično na strežnik prenašajo tudi zasebni ključi, kar je iz varnostnih razlogov daleč od optimalne rešitve.

Orodje OpenDNSSEC za generiranje in shranjevanje ključev ter podpisovanje zon podpira uporabo strojnih modulov HSM, združljivih z različnimi standardi FIPS 140-X. Ti moduli se lahko zaklenejo, kar pomeni, da napadalec ne more dostopati do ključev tudi ob neposrednem vdoru v sistem.

3.4 Zaščita pred izgubo podatkov

Če se zgodi, da se napadalec dokoplje do naših zasebnih ključev, je treba v skladu s standardom RFC 5011 [14] te ključe takoj preklicati, generirati nove in z njimi ponovno podpisati zono. Preklic ključev na ta način je mogoč v orodju BlueCat, preostali dve orodji pa slednje funkcije ne podpirata.

Če želimo ob izgubi podatkov ponovno vzpostaviti prejšnje stanje, moramo imeti varnostne kopije. Orodji BlueCat in Infoblox, ki vso konfiguracijo in ključe hranita v bazi podatkov, podpirata njihovo varnostno kopijo. Orodje Infoblox dodatno podpira izvoz posameznih ključev prek vmesnika API (angl. Application Programming Interface). Format izvožene datoteke je lastniški. BlueCat izvoza posameznih ključev prek vmesnika API pa ne omogoča.

Pri orodju OpenDNSSEC nimamo centralne baze podatkov, zato moramo poleg datotek z zonami varnostno kopirati tudi datoteke s konfiguracijo, bazo podatkov konfiguracije in repozitorije s ključi. Prav tako je ob delu z repozitorijem SoftHSM mogoč izvoz posameznih ključev v datoteko PEM, ki je združljiva s standardom PKCS#8 [16]. Če uporabljamo drugi HSM, pa je izvoz ključev odvisen od njega.

3.5 Prehod med različnimi orodji

Izvoz zon je relativno preprost, saj vsa orodja omogočajo repliciranje zon na druge strežnike DNS (Bind, NSD, Microsoft DNS). Glede izvoza ključev pa smo že omenili, da pri orodju BlueCat ni mogoč, pri orodju Infoblox pa so ključi izvoženi v lastniškem formatu. To pomeni, da je prehod s teh dveh orodij DNSSEC na tretje otežen. Orodji imata ključe sicer shranjene tudi v datotekah strežnika Bind. Če nam uspe uspešno prenesti te datoteke iz njih, postane prehod bistveno lažji.

OpenDNSSEC nam skupaj s programskim repozitorijem SoftHSM prinaša tudi orodje za migracijo med izvoženimi ključi v formatu PKCS#8 in datotekami strežnika Bind. Če uporabljamo drug varnostni modul HSM, sta izvoz in uvoz ključev odvisna od proizvajalca.

Prehod bi na splošno lahko izvedli na dva načina:

- z uvozom zasebnih ključev iz prvega orodja v drugo,
- z uvozom zapisa DNSKEY na prvem orodju v nepodpisano zono na drugem orodju.

Pri uvozu zasebnih ključev iz prvega orodja v drugo je prehod iz plačljivih orodij na OpenDNSSEC mogoč, če

[4]Sekundarni sistem ne prevzame vloge primarnega samodejno. Za preklop je potreben ročni poseg administratorja.

nam uspe iz njih izvoziti zasebne ključe. Prehod z OpenDNSSEC na orodja, ki jih ponuja trg, pa ni mogoč, saj v tem primeru ne moremo uvoziti zasebnih ključev. Prehod iz OpenDNSSEC na orodja strežnika Bind in nasprotno je precej preprost, medtem ko je prehod iz plačljivih orodij na orodja strežnika Bind odvisen od tega, ali nam bo iz teh uspelo prenesti datoteke Bind s ključi. Prehod z orodij Bind na orodja, ki jih ponuja trg, ni mogoč, prav tako ni mogoč prehod med plačljivima orodjema BlueCat in Infoblox.

Drugi način bi bil, da v nepodpisano zono dodamo zapis DNSKEY. Postopek prehoda je naslednji:

- Uporabljamo prvo orodje.
- V drugem orodju generiramo par ključev ZSK/KSK.
- V nepodpisano verzijo zone vstavimo zapise DNSKEY iz drugega orodja.
- Zono še nekaj časa podpisujemo s prvim orodjem.
- Ključe iz prvega orodja vstavimo v obliki zapisov DNSKEY v nepodpisano zono drugega orodja.
- Zono začnemo podpisovati z drugim orodjem.

Tako BlueCat kakor Infoblox ne omogočata vstavljanja zapisov DNSKEY v nepodpisano zono. OpenDNSSEC tak način sicer omogoča, vendar bi ga bilo treba uporabiti v navezi s katerim izmed drugih orodij.

4 PRIMERJAVA ORODIJ ZA VZDRŽEVANJE KLJUČEV IN ZON

4.1 Definiranje politik

Algoritmi, podprti z različnimi orodji, so predstavljeni v tabelah Tabela 1, Tabela 2 in Tabela 3. Kakor je razvidno iz tabel, OpenDNSSEC ne podpira algoritmov DSA, Infoblox ima podprte vse algoritme razen uporabe opt-out pri negativnih odgovorih, BlueCat pa poleg tega, da prav tako pri negativnih odgovorih ne podpira opt-out, pri uporabi NSEC3 ne uporablja salt, ne podpira NSEC3 pri novejšem algoritmu RSA/SHA-256 in ne podpira algoritma RSA/SHA-512.

Tabela 1: Algoritmi, podprti z različnimi orodji

Orodje	Podprti algoritmi
BlueCat	RSA/SHA-1, RSA/SHA-256, DSA
Infoblox	RSA/SHA-1, RSA/SHA-256, RSA/SHA-512, DSA
OpenDNSSEC	RSA/SHA-1, RSA/SHA-256, RSA/SHA-512

Tabela 2: Podpora NSEC, NSEC3

Orodje	Negativni odgovori	Uporaba salt	Opt-Out
BlueCat	NSEC, NSEC3[5]	Ne	Ne
Infoblox	NSEC, NSEC3	Da	Ne
OpenDNSSEC	NSEC, NSEC3	Da	Da

Tabela 3: Podpora algoritmov zapisa DS

Orodje	Zapis DS
BlueCat	SHA-1
Infoblox	SHA-1, SHA-2
OpenDNSSEC[6]	SHA-1, SHA-2

Orodja omogočajo kreiranje različnih politik in smiselno umestitev zon v njih. OpenDNSSEC in BlueCat zahtevata striktno umestitev zone v politiko, medtem ko ima Infoblox politiko vezano na pogled (angl. DNS view). Pri orodju Infoblox je mogoče parametre podpisovanja spremeniti na posamezni zoni[7].

Najprej smo želeli zone podpisati z algoritmom 8 (RSA/SHA-256) in za negativne odgovore uporabljati NSEC3, vendar se je izkazalo, da eno izmed orodij pri uporabi tega algoritma ne podpira uporabe NSEC3 za negativne odgovore. Posledično smo se odločili, da jih podpišemo z algoritmom 7 (RSA/SHA-1 NSEC3). Za podpisovanje zon smo uporabili naslednjo politiko:

- Algoritem: 7.
- Negativni odgovori: NSEC3 (brez Opt-Out).
- Velikost KSK: 2048 bitov.
- Velikost ZSK: 1024 bitov.
- Trajanje podpisov: 7 dni.
- Rok veljavnosti ključa ZSK: 11 dni.
- Rok veljavnosti ključa KSK: 14 dni.
- Način menjave ključa ZSK: predhodna objava ključa.
- Način menjave ključa KSK: dvojno podpisovanje.

4.2 Podpisovanje zon

Zone je bilo treba pred podpisovanjem uvoziti na posamezna orodja. Najlažji je bil uvoz v OpenDNSSEC. Izvesti ga je mogoče na več načinov, ki jih ponuja sistem Linux, na katerem ta teče. V testnem okolju smo jih uvozili prek protokola SSH. Za sistem Infoblox smo v ta namen napisali skripto, ki kliče funkcije API orodja. Skripta je sledila seznamu zon in za vsako izmed njih s strežnika DNS sprožila prenos (angl. zone transfer). Na BlueCat smo zone uvozili prek funkcije uvoza podatkov s pomočjo datoteke XML. Datoteko

[5] NSEC3 ni podprt ob uporabi RSA/SHA-256.
[6] DS vstavimo v starševsko zono ročno.
[7] Vsaka zona posebej ne potrebuje svoje kreirane politike.

XML smo pripravili iz Bindovih datotek zon s pomočjo internega orodja, ki nam ga je posredoval BlueCat.

Podpisovanje na OpenDNSSEC smo sprožili tako, da smo zono dodali v seznam zon in ji dodelili ustrezno politiko. Na sistemih Infoblox in BlueCat je treba za vsako zono določiti, ali uporablja DNSSEC. Na sistemu Infoblox smo znova uporabili skripto, ki je sledila seznamu, in za vsako zono sistemu sporočila, naj jo podpiše. Pri orodju BlueCat se je na tem mestu zapletlo, saj je mogoče podpisati zgolj posamično zono, in še to samo prek grafičnega vmesnika (BlueCat API ne podpira opravil v zvezi z DNSSEC). Tako nam ni preostalo drugega, kot da smo to naredili ročno, prek grafičnega vmesnika. Časi generiranja 401 2048-bitnih ključev KSK in 401 1024-bitnih ključev ZSK so predstavljeni v Tabela 4.

Tabela 4: Čas generiranja ključev

Orodje	Generiranje 401 ključev	
	2048-bit KSK	1024-bit ZSK
BlueCat	16 min	2 min
Infoblox	ni podatka	ni podatka
OpenDNSSEC	9 min	1 min 30 s

Pri orodju Infoblox časov nismo mogli oceniti, saj so se ključi generirali hkrati s podpisovanjem posamične zone. Pri OpenDNSSEC so se ključi začeli generirati, ko smo zagnali servis. Pri BlueCat so se ključi prav tako generirali takoj, ko smo za posamično zono določili, da uporablja DNSSEC, vendar smo čase lahko izmerili naknadno ob ponovnem generiranju ključev v času menjave ključev. Takrat so se ključi generirali pred začetkom podpisovanja zon.

Pomemben parameter pri generiranju ključev je tudi možnost njihovega vnaprejšnjega generiranja. Rešitev nam zlasti pride prav, ko želimo zakleniti HSM. Po zaklepu pisanje po varnostnem modulu namreč ni več možno. Prav tako obstaja opcija, da ključe generiramovnaprej in jih prenesemo na orodje, ki je v pripravljenosti. Po prenosu ključev na redundantno lokacijo varnostni modul HSM zaklenemo. V kolikor ne uporabljamo posebnega FIPS-združljivega varnostnega modula, je generiranje ključev vnaprej varnostno precej tvegano opravilo. Napadalec se ob vdoru dokoplje tudi do ključev, ki jih bomo uporabljali v prihodnosti. Generiranje ključev vnaprej nam omogoča OpenDNSSEC, prav tako je OpenDNSSEC edina rešitev, ki nam omogoča skupno rabo ključev med zonami. Skratka, več zon je lahko podpisanih z istimi ključi.

Ko smo ključe uspešno generirali, smo se lotili podpisovanja zon. Čas trajanja prvega in ponovnega podpisovanja 401 zon se nahaja v Tabela 5.

Tabela 5: Čas podpisovanja zon

Orodje	Prvo podpisovanje 401 zon	Ponovno podpisovanje 401 zon
BlueCat	1 h	1 h 11 min
Infoblox	ni podatka[8]	ni podatka
OpenDNSSEC	3 h 17 min	3 h 11 min
OpenDNSSEC brez orodja Auditor	1 h 55 min	2 h 14 min

BlueCat in Infoblox podpisujeta zone z Bindovimi orodji. OpenDNSSEC ima podpisovanje zon samostojno implementirano, za to opravilo ne uporablja zunanjih orodij. Trajanje podpisovanja v OpenDNSSEC je daljše tudi zato, ker OpenDNSSEC ponuja tudi orodje Auditor, ki po končanem podpisovanju preveri, ali so zone pravilno podpisane. Šele ko je zona uspešno preverjena, se namesti v strežnik DNS. BlueCat in Infoblox podpisanih zon ne preverjata dodatno, sicer pa zone preveri strežnik Bind, preden jih naloži. Kljub temu dodatno večkratno preverjanje pravilnosti podpisov zon ne pomeni pomanjkljivosti.

Ob podpisovanju zone je pomembno, da se starševski zoni posreduje zapis DS ključa KSK. Ker smo imeli starševsko zono dnssec.si in njene poddomene naložene na istem sistemu, bi lahko bilo posredovanje zapisov DS poddomen starševski zoni dnssec.si avtomatizirano. Orodji BlueCat in Infoblox sta zapise DS tudi posredovali na ta način, medtem ko OpenDNSSEC tega ni izvajal. OpenDNSSEC sicer omogoča uporabo skript, prek katerih bi lahko samodejno posredovali zapise DS npr. prek sistema EPP (angl. Extensible Provisioning Protocol).

Testi so pokazali, da ponovno podpisovanje zon traja približno enako dolgo kakor prvo podpisovanje. Pri orodjih BlueCat in Infoblox za ponovno podpisovanje skrbi kar strežnik Bind, OpenDNSSEC pa izvaja to operacijo sam.

4.3 Menjava ključev

Orodje OpenDNSSEC menjava ključ ZSK po metodi predhodne objave ključa, ključ KSK pa z metodo dvojnega podpisovanja. Infoblox pri menjavah obeh ključev uporablja metodo dvojnega podpisovanja, BlueCat pa uporabniku na tem mestu omogoča, da sam določi, na kakšen način želi zamenjati ključe. Vsa orodja pred menjavo ključev te tudi generirajo. OpenDNSSEC jih ne generira, če jih ima generirane že vnaprej.

Menjava ključev je potekala v skladu z definiranimi politikami v 4.1. Orodje BlueCat pri menjavi ključa po načinu predhodne menjave ključa tega najprej zapiše v

8 Naši testni podatki so presegali omejitev velikosti baze orodja Infoblox, zato časov pri tem orodju nismo mogli izmeriti.

zono in se podpiše s KSK. Zona se ne podpisuje ponovno, zato je taka operacija trajala zgolj nekaj minut. Ob naslednjem časovnem intervalu se zona podpiše z novim ključem ZSK, stari pa ostane v zoni, zato je bila ta operacija časovno ekvivalentna operaciji podpisovanja zone. Ob menjavi ključa KSK po načinu dvojnega podpisovanja pa se je poleg ključa ZSK znova podpisala tudi zona z obstoječim ključem ZSK, kar je bila nepotrebna in časovno potratna operacija. Pri menjavi ključa KSK se je ustrezno spreminjal tudi zapis DS v starševski zoni. Paziti moramo samo pri posodabljanju zapisov DS v zonah, ki se ne upravljajo na orodju. Ključi se namreč samodejno zamenjajo, ne glede na to, ali so ustrezno posredovani in objavljeni v starševski zoni. Nismo bili obveščeni, da je treba zamenjati zapis DS prek grafičnega vmesnika.

OpenDNSSEC je ključe ZSK zamenjal samodejno, ključev KSK pa ne bo nikoli zamenjal samodejno. Administrator bo menjavo ključev KSK moral sprožiti ročno.

Pri orodju Infoblox se ključ ZSK prav tako zamenja samodejno, menjavo ključa KSK pa mora administrator sprožiti ročno. Ko se bliža čas menjave ključa, dobi administrator v grafičnem vmesniku obvestilo in možnost, da sproži menjavo ključa KSK. Zapisi DS v zonah na lokalnem orodju se samodejno posodabljajo, zonam zunaj našega administrativnega območja pa moramo sami sporočiti nove zapise.

Menjava algoritmov pri orodju OpenDNSSEC ni mogoča. Na orodju Infoblox je algoritme mogoče zamenjati, ni pa mogoč prehod iz NSEC na NSEC3 in nasprotno. BlueCat omogoča prehod med NSEC in NSEC3.

OpenDNSSEC zone podpisuje pravilno, le proces OpenDNSSEC je nestabilen. Če mu zmanjka pomnilnika RAM, se zruši (medpomnilnika na disku ne uporablja). Sicer so v dani zahtevnosti testov vsa orodja imela težave s sistemskimi viri.

5 UPORABNIŠKE IZKUŠNJE

Uporaba obeh plačljivih orodij je z grafičnim vmesnikom zelo preprosta. Za vzpostavitev delovanja DNSSEC ni treba imeti tako rekoč nikakršnega predznanja. Na orodju Infoblox se DNSSEC vklopi z enim klikom, na BlueCat je treba sicer prej definirati politiko, ki se pozneje poveže z zono (potreben je torej en korak več). Velja opomniti, da je pri obeh orodjih privzeta politika nastavljena na algoritem RSA/SHA-1 NSEC, kar pomeni, da uporaba privzetih nastavitev ni najbolj varna zaradi možnosti posrednega prenosa celotne zone (t. i. zone walking). Dobro je poskrbljeno za spremembo privzetih nastavitev in politik na preprost način. Obe orodji imata vgrajen odličen vmesnik za pomoč pri delu, prav tako sta dobro dokumentirani. Pri orodju BlueCat sicer ni mogoče pregledovati podpisanih zon, saj na sistemu Proteus, do katerega dostopamo prek grafičnega vmesnika, zone niso podpisane (podpisujejo se šele na strežniku Adonis). Po drugi strani je delo z OpenDNSSEC zapleteno, potrebnega je precej znanja o DNSSEC, orodje pa si moramo tudi sami namestiti. Imamo pa prednost, saj lahko nastavljamo bolj specifične parametre, ki jih plačljiva orodja ne omogočajo.

Z obema plačljivima orodjema je mogoče delo prek API, ki smo se ga posluževali zlasti za avtomatizacijo ponavljajočih se opravil. BlueCat API uporablja standardiziran dostop prek SOAP (angl. Simple Object Access Protocol), žal pa ne podpira vseh funkcionalnosti, ki jih ponuja orodje prek grafičnega vmesnika. Infoblox po drugi strani prek API omogoča vsa opravila, ki jih podpira. OpenDNSSEC za zdaj še ne podpira dela prek API, vendar v primeru njegove uporabe ni nujen pogoj, saj nismo omejeni z grafičnim vmesnikom, v ukazni lupini sistema Linux pa smo lahko avtomatizirali marsikatero naše opravilo.

Pri plačljivih orodjih je slabše poskrbljeno za odpravljanje napak. Imamo sicer dostop do dnevniških datotek, vendar je pregledovanje v grafičnem vmesniku zaradi prevelike količine podatkov precej oteženo. BlueCat nam omogoča prenos dnevniških datotek iz strežnika, vendar ne na sistemu Adonis, kjer se je tudi težko dokopati do starejših dnevniških datotek. Beleženje v dnevniške datoteke s strani OpenDNSSEC je precej podrobno in možnost odpravljanja napak je velika. Plačljivi orodji podpirata obveščanje uporabnika prek protokola SNMP (angl. Simple Network Management Protocol) in elektronske pošte. Tako uporabnika prek teh protokolov obvestita npr., kdaj bo izvedena menjava ključev.

Pri testih smo uživali izredno dobro in kvalitetno podporo s strani BlueCat Networks, podjetje Infoblox nam je prav tako zagotovilo nekaj podpore. Glede na to, da smo podporo dobivali že pri samem testiranju, je najverjetneje podpora za stranke še toliko boljša. Pri OpenDNSSEC je tako kot pri vseh odprtokodnih orodjih podpora na voljo v različnih odprtokodnih skupnostih (e-poštni seznami, forumi).

6 SKLEP

Za lažje zagotavljanje kakovostnih storitev DNSSEC ponudnikom storitev DNS, implementaciji katerih se v prihodnosti ne bodo mogli izogniti, smo testirali dve plačljivi in eno odprtokodno orodje za vzdrževanje zon in ključev. Na eni veliki in 400 srednje velikih zonah smo testirali njihove tehnične značilnosti, kot so podpora standardom DNSSEC, hitrost in zanesljivost delovanja, varnost in visoka razpoložljivost. Preverili smo tudi zahtevnost in prijaznost uporabe.

Za enega izmed orodij je bil glede na omejitve testne licence test hitrosti delovanja prezahteven, preostali orodji pa sta se kljub velikim omejitvam sistemskih virov dobro obnesli. Orodja imajo implementirano večino standardov DNSSEC. Vsa orodja lahko zanesljivo delujejo v visokorazpoložljivem načinu, manjka pa možnost vključitve zunanjih orodij za preverjanje pravilnosti podpisovanja zon. Glede varnosti

smo ugotovili, da pri orodjih na trgu manjka podpora standardom FIPS. Plačljivi orodji sta lažji za uporabo in imata uporabniški vmesnik, ki je uporabniku bolj prijazen kot tisti pri odprtokodnem orodju.

Ker težave pri delovanju sistema DNSSEC pomenijo, da domene postanejo v internetu nedosegljive, bomo svoje delo nadaljevali z analizo in uvajanjem orodij, ki bodo preverjala ustreznost podpisanih zon in napake odkrila, še preden bo zona objavljena na internetu.

ZAHVALA

Posebna zahvala Freyu Khademiju iz BlueCat Networks za izredno podporo pri testiranju njihovega DNSSEC orodja. Prav tako gre zahvala za tehnično pomoč tudi podjetju Infoblox. Obema podjetjema smo tudi hvaležni, da so nam omogočili testiranje svojih DNSSEC orodij.

LITERATURA

[1] P. Albitz, C. Liu, *DNS and Bind*, Sebastopol: O'Reilly Media, 2006, pogl. 11.

[2] A Review of Administrative Tools for DNSSEC – Spring 2010, http://www.iis.se/docs/DNSSEC-Admin-tools-review-Final.pdf, dostopno avgusta 2011.

[3] Bind Documentation, http://www.isc.org/software/bind/documentation, dostopno avgusta 2011.

[4] BlueCat IPAM, http://www.bluecatnetworks.com/, dostopno avgusta 2011.

[5] FIPS Publications, http://csrc.nist.gov/publications/PubsFIPS.html, dostopno avgusta 2011.

[6] C. Florent, Security Issues with DNS, SANS Institute, 2003, http://www.sans.org/reading_room/whitepapers/dns/security-issues-dns_1069, dostopno julija 2011.

[7] Infoblox Administrator Guide, http://ww2.infoblox.com/support/tech_lib/NIOS/NIOS_AdminGuide_5.1r2.pdf, dostopno avgusta 2011.

[8] KOS, Andrej, BEŠTER, Janez. Evolucija hrbteničnih IP-omrežij v smeri MPLS. Elektrotehniški vestnik. [Slovenska tiskana izd.], 2001, letn. 68, št. 4, str. 200–206. [COBISS.SI-ID 2505556]

[9] KOS, Andrej, BEŠTER, Janez. Razvoj in uvajanje novih telekomunikacijskih storitev. Elektrotehniški vestnik. [Slovenska tiskana izd.], 2002, letn. 69, št. 3-4, str. 221–226. [COBISS.SI-ID 3318100]

[10] OpenDNSSEC, OpenDNSSEC Documentation, http://www.opendnssec.org/documentation/, dostopno avgusta 2011.

[11] RFC 4033, http://www.rfc-archive.org/getrfc.php?rfc=4033, dostopno julija 2011.

[12] RFC 4034, http://www.rfc-archive.org/getrfc.php?rfc=4034, dostopno julija 2011.

[13] RFC 4035, http://www.rfc-archive.org/getrfc.php?rfc=4035, dostopno julija 2011.

[14] RFC 5011, http://www.rfc-archive.org/getrfc.php?rfc=5011, dostopno avgusta 2011.

[15] RFC 5155, http://www.rfc-archive.org/getrfc.php?rfc=5155, dostopno julija 2011.

[16] RFC 5208, http://www.faqs.org/rfcs/rfc5208.html, dostopno avgusta 2011.

[17] SEDLAR, Urban, ZEBEC, Luka, BEŠTER, Janez, KOS, Andrej. Bringing click-to-dial functionality to IPTV users. IEEE commun. mag. (Print). [Print ed.], Mar. 2008, vol. 46, no. 3, str. 118–125, ilustr. [COBISS.SI-ID 6398548]

[18] STEGEL, Tine, STERLE, Janez, SEDLAR, Urban, BEŠTER, Janez, KOS, Andrej. SCTP multihoming provisioning in converged IP-based multimedia environment. Comput. commun.. [Print ed.], 2010, vol. 33, no. 14, str. 1725–1735, ilustr. [COBISS.SI-ID 7960404]

[19] ŠTERN, Andrej, KOS, Andrej. Mobilni telefon kot orodje na področjih varovanja zdravja = Mobile phone as a tool in the areas of health protection. Zdrav Vestn (Tisk. izd.). [Tiskana izd.], nov. 2009, letn. 78, št. 11, str. 673–684, ilustr. [COBISS.SI-ID 7486548]

[20] UMBERGER, Mark, HUMAR, Iztok, KOS, Andrej, GUNA, Jože, ŽEMVA, Andrej, BEŠTER, Janez. The integration of home-automation and IPTV system and services. Comput. stand. interfaces. [Print ed.], Jun. 2009, vol. 31, no. 4, str. 675–684, ilustr. [COBISS.SI-ID 7093332]

[21] VOLK, Mojca, GUNA, Jože, KOS, Andrej, BEŠTER, Janez. Quality-assured provisioning of IPTV services within the NGN environment. IEEE commun. mag. (Print). [Print ed.], May 2008, vol. 46, no. 5, str. 118–126, ilustr. [COBISS.SI-ID 6532436]

[22] ZEBEC, Luka, HUMAR, Iztok, BODNARUK, Darko, KOS, Andrej, BEŠTER, Janez. NGN service development - overview and Parlay X implementation. Elektrotehniški vestnik. [Slovenska tiskana izd.], 2005, letn. 72, št. 1, str. 45–51, ilustr. [COBISS.SI-ID 4771156]

Dušan Kozic je diplomiral leta 2011 na Fakulteti za računalništvo in informatiko v Ljubljani. Zaposlen je kot upravljavec sistema DNS na Arnesu. Njegova raziskovalna zanimanja vključujejo področje omrežne varnosti.

Benjamin Zwittnig je diplomiral leta 1990 na Fakulteti za naravoslovje in tehnologijo – Fakulteta za matematiko. Na Arnesu je zaposlen od leta 1992. V tem času je med drugim skrbel za tehnični razvoj registracije domen in s tem povezano upravljanje vrhnjih DNS strežnikov za .SI.

Janez Sterle je diplomiral leta 2003 na Fakulteti za elektrotehniko Univerze v Ljubljani. Zaposlen je v Laboratoriju za telekomunikacije (LTFE) na Fakulteti za elektrotehniko. Njegova raziskovalna področja so internetni protokol naslednje generacije, omrežna varnost, prometno načrtovanje in prometna analiza ter razvoj in vpeljava integriranih storitev v fiksna in mobilna omrežja naslednje generacije.

Andrej Kos je doktoriral leta 2003 na Fakulteti za elektrotehniko Univerze v Ljubljani in bil istega leta izvoljen v naziv docent za področje elektrotehnike ter v letu 2009 v naziv izredni profesor. Zaposlen je kot učitelj in raziskovalec, predava predmete s področij telekomunikacij in večpredstavnosti. V okviru znanstveno-raziskovalnega dela se posveča telekomunikacijskim, multimedijskim in internetnim omrežjem ter sistemom na dostopovnem, agregacijskem in hrbteničnem sloju, testiranju, prometnim analizam in optimizaciji virov, krmilnim protokolom ter razvoju konvergenčnih multimedijskih storitev.