Helmut Reiser*, Daniel Feuchtinger, Wolfgang Hommel, Bernhard Schmidt und Michael Storz

DNSSEC – Konzepte und Betriebsaspekte des Domain Name Systems der Zukunft

DOI 10.1515/pik-2015-0005

Abstract: Das Domain Name System (DNS) gehört zu den elementarsten Mechanismen des Internet. Es ermöglicht nicht nur die bequeme Nutzung des World Wide Web (WWW) durch die Abbildung gut merkbarer Rechnernamen auf IP-Adressen, sondern stellt eine Reihe weiterer Daten zur Verfügung, ohne die beispielsweise der E-Mail-Versand über das Internet nicht funktionieren würde. Wie anderen Protokollen aus den Gründerzeiten des Internet mangelt es DNS jedoch an Sicherheitseigenschaften. DNSSEC tritt an, die dringendsten Defizite zu beseitigen, doch der verfolgte Ansatz ist nicht unumstritten. In diesem Artikel werden zunächst die konzeptionellen Grundlagen, die Fortschritte bei der Entwicklung in den letzten zehn Jahren und die Vorteile sowie die möglichen Nachteile der wesentlichen DNSSEC-Bestandteile dargelegt. Anschließend wird auf betriebliche Herausforderungen und praktischen Mehrwert des DNSSEC-Einsatzes am Beispiel des vom Leibniz-Rechenzentrum betriebenen Münchner Wissenschaftsnetzes eingegangen.

Motivation

Jeder Internet-Nutzer ist intuitiv mit dem Konzept von Domains vertraut: Wer die URLs seiner bevorzugten sozialen Netzwerke, Presseorgane oder anderer Internet-Dienste im Webbrowser eintippt, setzt sich implizit damit auseinander, dass unterschiedliche Top-Level-Domains wie „.com“ oder „.de“ existieren und der Namensteil „www.“ auf einen Webserver hinweist. Allerdings beschäftigen sich nur die technisch Interessierteren damit, dass vor den Anwendern verborgen im Hintergrund das Domain Name System (DNS) arbeitet, um beispielsweise die gut merkbaren Rechnernamen wie „www.facebook.com“ auf IP-Adressen abzubilden, auf deren Basis die vernetzte Kommunikation im gesamten Internet funktioniert. DNS leistet jedoch noch wesentlich mehr, da es letztlich eine global verteilte hierarchische Datenbasis implementiert, über die nahezu beliebige Informationen weltweit abrufbar hinterlegt werden können. Andere Protokolle hängen essentiell von einem funktionierenden DNS ab; beispielsweise würde der weltweite E-Mail-Versand über das SMTP-Protokoll ohne die im DNS gespeicherten Mail-Exchange-Informationen (MX-Records) nicht funktionieren.

Die massive Abhängigkeit nahezu sämtlicher Internet-Anwendungen von DNS impliziert nicht nur, dass der von DNS-Servern bereitgestellte Dienst hohe Anforderungen an seine Verfügbarkeit erfüllen muss, sondern dass die über DNS bezogenen Informationen auch eine hohe Datenqualität aufweisen müssen. Diese Schlüsseleigenschaft zeigt sich sowohl bei Schadsoftware wie „Banking-Trojanern“ als auch bei Diskussionen über staatlich angeordnete Internet-Zensur: Wer DNS-Informationen manipulieren kann, entscheidet praktisch darüber, auf welche Internet-Dienste ein Anwender zugreifen kann und ob er sich z. B. mit dem richtigen Webserver seiner Bank verbindet oder mit der von Kriminellen nachgebauten Banking-Webseite.

DNS, das als Internet-Protokoll 1984 das Licht der Welt erblickte RFC 920, hat seit jeher organisatorische und technische Aspekte. Organisatorisch muss beispielsweise festgelegt werden, welche Top-Level-Domains existieren, wer sie verwaltet und welche natürlichen oder juristischen Personen welche generischen oder länderspezifischen Domains registrieren dürfen. Die technische Seite betrifft

- die server- und client-seitig eingesetzten Softwarekomponenten;
- das DNS-Protokoll, mit dem die Kommunikation unter den Softwarekomponenten erfolgt; sowie
- die Festlegung der Rollenverteilung bei der Kommunikation.

Letztere ist für das Zusammenspiel der einzelnen Server ausschlaggebend:

- Root Nameserver verwalten die Wurzel des globalen, hierarchischen DNS-Namensraums. Sie erteilen beispielsweise Auskunft darüber, welche anderen Nameserver für die Top-Level-Domain „.de“ zuständig sind. Logisch gesehen existieren weltweit nur 13 Root Nameserver, die von 12 verschiedenen Organisationen

*** Kontaktperson: Helmut Reiser:** E-Mail: helmut.reiser@lrz.de
Daniel Feuchtinger: E-Mail: Daniel.Feuchtinger@lrz.de
Wolfgang Hommel: E-Mail: wolfgang.hommel@lrz.de
Bernhard Schmidt: E-Mail: bernhard.schmidt@lrz.de
Michael Storz: E-Mail: Michael.Storz@lrz.de

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

betrieben werden; physisch stecken dahinter Tausende von Servern, die weltweit auf hunderte Rechenzentren verteilt und per Anycast-Routing Internet-topologisch optimiert erreichbar sind. Die IP-Adresse mindestens eines Root-Servers muss für die Namensauflösung bekannt sein; es gibt keine integrierte Möglichkeit, sie über DNS herauszufinden.

– Für eine Domain originär zuständige DNS-Server werden als autoritative Nameserver bezeichnet. Dabei wird betrieblich zwischen genau einem Primary Nameserver und einem bzw. mehreren Secondary Nameservern unterschieden. Die Secondary Nameserver dienen der Ausfallsicherheit und Lastverteilung, wobei pro Top-Level-Domain organisatorisch vorgegeben wird, wie viele Secondary Nameserver mindestens vorhanden sein müssen – in der Regel einer oder zwei.
– Die Namensauflösung für Endgeräte übernehmen so genannte DNS-Resolver, die von Anwendungen bzw. dem Betriebssystem zur Ermittlung von DNS-Informationen genutzt werden. Üblicherweise arbeitet in den Betriebssystemen von IP-fähigen Endgeräten ein sogenannter Stub-Resolver mit reduzierten Fähigkeiten, der die Anfrage an einen richtigen Resolver weiterleitet, dessen IP-Adresse manuell oder per DHCP auf dem Endgerät konfiguriert wurde. Diese als lokale oder Caching-only Nameserver bezeichneten Resolver übernehmen die iterativen Anfragen im DNS-Baum ausgehend von den Root-Servern und cachen für jede Ebene die Ergebnisse entsprechend deren Time-To-Live-Angabe (TTL), was insbesondere in den oberen Ebenen (Root-Zone und Top-Level-Domains) die autoritativen Server bei nachfolgenden Anfragen durch denselben oder andere Clients signifikant entlastet.

Zur Auflösung des Rechnernamens von „www.facebook.com“ müssen, sofern noch keine Einträge in Caches vorliegen, ein Root Nameserver, ein für „.com“ zuständiger Nameserver und schließlich ein für „facebook.com“ autoritative Nameserver vom Resolver angefragt werden, bis letzterer nach IP-Adressen für den Hostnamen „www“ befragt werden kann. Das zur Einzelabfrage von Resource Records (RRs) verwendete DNS-Protokoll ist UDP-basiert (Standard-Port 53/UDP). Eine zentrale Schwachstelle des herkömmlichen DNS liegt darin, dass den UDP-Antwortpaketen auf DNS-Anfragen nicht anzusehen ist, ob sie wirklich vom befragten DNS-Server stammen oder ob sie von einem Angreifer kommen, der die Absender-IP-Adresse des entsprechenden UDP/IP-Pakets mit trivialen Mitteln gefälscht hat. Im letzteren Fall liegt so genanntes DNS-Spoofing vor, da es dem Angreifer beispielsweise gelingt, auf eine Anfrage nach „www.facebook.com“ mit der IP-Adresse seines eigenen Webservers statt der des richtigen Facebook-Webservers zu antworten. Wird die gefälschte Antwort nicht nur einem beliebigen Endgerät untergeschoben, sondern einem DNS-Resolver, ergibt sich ein Cache Poisoning, das dazu führt, dass die gefälschte Antwort für den Zeitraum der ebenfalls manipulierten TTL auch noch an beliebig viele andere anfragende (Stub-)Resolver weitergegeben wird. Cache-Poisoning ist auch auf Ebene der Top-Level-Domains möglich, so dass durch einen falschen Delegations-Record beispielsweise der de-Zone sämtliche de-Domains nach Belieben umgeleitet werden können.

Gelingt es einem Angreifer, den Netzverkehr an einer beliebigen Stelle zwischen Client und Root-Server zu kontrollieren, so sind DNS-Angriffe noch einfacher und können durch Umleiten des DNS-Verkehrs auf IP-Ebene manipuliert werden. Die einfachste Möglichkeit dieser Art besteht vielerorts darin, dem Angriffsziel per DHCP einen kompromittierten Resolver oder ein kompromittiertes Gateway zu konfigurieren. Auch die Kommunikation mit den Root-Servern ist weniger sicher als oft angenommen, da in der Regel nicht klar ist, wohin die Route zum nächsten Root-Server vom aktuellen Standort aus zu einem bestimmten Zeitpunkt verläuft und wer dort den DNS-Server betreibt.

Angriffe dieser Art wurden bis Mitte 2008 durch unzureichend abgesicherte DNS-Server-Implementierungen, auf die der Sicherheitsforscher Dan Kaminsky medienwirksam hingewiesen hat DNSKam, zusätzlich begünstigt, sind aber protokollinhärent und existieren somit nach wie vor. In der organisationsinternen Risikobewertung spielen sie oftmals keine große praktische Rolle, da die Betreiber von zwangsweise anfälligen DNS-Servern gerne davon ausgehen, dass ihr lokales Netz und der unmittelbare Internet-Uplink ausreichend sicher und vertrauenswürdig sind, so dass es Angreifern nicht gelingt, ausgehende DNS-Anfragen mitzuhören und gefälschte Antwortpakete noch vor der Antwort des richtigen angefragten Servers einzuschleusen oder den DNS-Verkehr auf IP-Ebene zu beeinflussen. Auf Situationen, die sich durch kompromittierte Systeme im lokalen Netz oder durch den Aufenthalt von Clients in weniger vertrauenswürdigen Netzen, beispielsweise bei Auslandsaufenthalten reisender Wissenschaftler, ergeben können, wird dabei oft nur unzureichend oder überhaupt nicht eingegangen.

DNSSEC RFC 4033, RFC 4034, RFC 4035 zielt als sicherheitsspezifische Erweiterung von DNS deshalb primär darauf ab, die Integrität und Authentizität von DNS-Informationen sicherzustellen; anders als vielleicht intuitiv zu erwarten spielt die Transport- oder Ende-zu-Ende-Verschlüsselung von DNS-Kommunikation bei DNSSEC keine Rolle. Im nächsten Abschnitt wird der aktuelle Stand der Technik hinter DNSSEC beschrieben. Da DNSSEC nicht nur

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

Beifall geerntet hat, werden anschließend Kritikpunkte diskutiert und betrachtet, wie sich DNSSEC in den letzten zehn Jahren diesbezüglich weiterentwickelt hat. Dass DNSSEC bereits durchaus praxistauglich ist, aber mit zahlreichen betrieblichen Stolperfallen einhergeht, wird nachfolgend am Beispiel des Leibniz-Rechenzentrum (LRZ), das im Münchner Wissenschaftsnetz (MWN) seit einiger Zeit mehr als 180.000 Endgeräte mit DNSSEC zwangsbeglückt, geschildert. Abschließend werden der praktische Mehrwert, der sich auch für andere Dienste durch DNSSEC ergibt, dargestellt und ein Fazit mit Handlungsempfehlungen gezogen.

Technik

Die geschilderten Angriffe haben die Eigenschaft gemeinsam, dass sie die fehlende Authentizität von DNS-Informationen ausnutzen. DNSSEC löst dieses Problem durch den Einsatz von asymmetrischer Kryptographie (in der Regel RSA/SHA256 (weitere siehe iana-dnssec-alg), mit der im DNS hinterlegte Informationen digital signiert werden. Die Signaturen sind für alle Teilnehmer am DNS validierbar, wodurch DNS-Angriffe durch Spoofing wirkungsvoll verhindert werden. DNSSEC bietet dabei die vollständige Absicherung der DNS-Inhalte und den Beweis der Existenz oder auch der Nicht-Existenz eines bestimmten Eintrags.

Wie jede aktuelle Anwendung asymmetrischer Kryptographie basiert auch DNSSEC auf der Erstellung von Schlüsselpaaren, die einen privaten und einen öffentlichen Teil haben. Der private Schlüssel muss vom Administrator sicher verwahrt werden und dient zur Erstellung der Signaturen, während der öffentliche Schlüssel verbreitet wird und zur Verifikation der Signaturen herangezogen werden kann. Der öffentliche Schlüssel wird dazu zusammen mit anderen Informationen, wie dem verwendeten Signaturverfahren, in der Zone selbst in einem DNSKEY-Record veröffentlicht. Die Schlüsselpaare werden jedoch direkt verwendet, sind also keine Zertifikate, wie sie von X.509v3 RFC 5280 bekannt sind, und enthalten daher keine Informationen über Vertrauensstellungen, Gültigkeitsdauer oder Inhaber.

Mithilfe des privaten Schlüssels werden nun alle Einträge der Zone signiert, indem für jede Kombination aus Name (*Label*), Typ und verwendetem Key eine Signatur erzeugt und diese in einem RRSIG-Record (Resource-Record-Signature) in der Zone hinterlegt wird. Die RRSIG-Records enthalten neben der Signatur zusätzliche Informationen, unter anderem das verwendete Signaturverfahren, eine Key-ID zur Unterscheidung mehrerer Signaturen bzw. Schlüssel und die Gültigkeitsdauer der Signatur. Die Gültigkeitsdauer kann vom Betreiber der Zone in weiten Bereichen selbst bestimmt werden und ist eine Abwägung aus dem Aufwand zur Neusignierung auf der einen Seite und der Anfälligkeit für Replay-Angriffe auf der anderen Seite. Mit Hilfe der RRSIG-Records können alle Einträge in einer DNS-Zone validiert werden.

Es gibt jedoch noch einen zusätzlichen, nicht ganz offensichtlichen Angriffsvektor, der geschlossen werden muss. Nicht nur im DNS hinterlegte Informationen können ein potentielles Sicherheitsproblem auslösen, auch eine vorgetäuschte Nicht-Existenz (*NXDOMAIN*/*NODATA*) eines bestimmten Eintrags kann Sicherheitsimplikationen haben, die von einem einfachen Denial-of-Service-Angriff bis hin zu Downgrade-Angriffen reichen. In einem naiven Design könnte der antwortende Nameserver eine spezifische „Existiert-nicht"-Antwort dynamisch generieren, mit seinem privaten Schlüssel signieren und dem Anfragenden zur Verfügung stellen. Der Aufwand einer Signaturerstellung ist jedoch um Größenordnungen höher als die normale DNS-Anfrage, so dass auch dies wieder einen trivialen Denial-of-Service (DoS) Angriffsvektor darstellen würde. Eine Vorabgenerierung aller möglichen angefragten Namen ist aufgrund des sehr großen Namensraum ebenfalls nicht möglich.

Um diesen Angriffsvektor zu verhindern, wurde bei der Spezifikation von DNSSEC auf einen cleveren Workaround gesetzt. Die existierenden Namen innerhalb einer Zone werden alphabetisch sortiert und über eine einfach verkettete Liste miteinander verbunden. Diese Informationen werden durch sogenannte NSEC-Records (Next SECure) ebenfalls in der Zone hinterlegt und signiert. Durch diese Verkettung existierender Namen können durch wenige NSEC-Records (mit den entsprechenden Signaturen) große Bereiche von nicht-existierenden Namen ausgeschlossen werden. Die NSEC-Records enthalten zusätzlich noch die für einen bestimmten Namen vorhandenen Typen, so dass auch die Existenz (oder Nicht-Existenz) eines bestimmten Typs bewiesen wird.

Das NSEC-Verfahren hat jedoch den großen Nachteil, durch die einfach verkettete Liste bei einem bekannten Startpunkt (dem Zonennamen) eine Auflistung aller Einträge in einer DNS-Zone zu erlauben (Zone-Walking), analog zu einem Zonentransfer über das Protokoll AXFR (Authoritative Transfer) RFC 5936, das üblicherweise zwischen Secondary und Primary Nameserver zum Einsatz kommt. Geheime (aber öffentlich auflösbare) Namen zu verwenden bringt zwar keine zusätzliche Sicherheit (Security-by-Obscurity, siehe auch dns-axfr), erschwert aber zumindest automatisierte Angriffe. Als Abhilfe gegen das Zone-Walking wurde schon vor Jahren der erweiterte Standard NSEC3 spezifiziert, bei dem von den Namen vor der Verkettung mit

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

einem frei definierbaren Seed und einer wählbaren Rundenanzahl ein kryptographischer Hash berechnet wird. Da vom Hash nicht auf den Namen rückgeschlossen werden kann, von einem angefragten Namen auf den Hash und damit die (Nicht-)Existenz in der NSEC3-Chain jedoch schon, verhindert dies effektiv ein Auflisten der Zone.

Die beschrieben Einträge ermöglichen die Validierung aller DNS-Einträge einer Zone, sofern der öffentliche Schlüssel als vertrauenswürdig konfiguriert ist. Dies skaliert jedoch bei Millionen Domains weltweit nicht, weswegen die Vertrauensstellung implizit durch Delegation erfolgen muss.

Die Public-Key-Infrastructure (PKI) für DNSSEC ist analog zum DNS-Baum aufgebaut: Der initiale Vertrauensanker besteht aus den öffentlichen Schlüsseln der Root-Zone, die zusammen mit den IP-Adressen der Root-Server auf jedem Resolver konfiguriert sein müssen. Von der Root-Zone werden die öffentlichen Schlüssel der Top-Level-Domains delegiert, welche wiederum die Schlüssel der untergeordneten Domains delegieren usw. Um den Delegationsmechanismus verstehen zu können, müssen zunächst die verwendeten Schlüsselarten bekannt sein. An die Schlüsselpaare für DNSSEC gibt es widersprüchliche Anforderungen: Einerseits soll es leicht möglich sein, die Schlüssel für eine Zone zu wechseln, andererseits soll die Delegationskette möglichst konstant bleiben und nicht jede Schlüsseländerung eine Änderung in den delegierenden Zonen zur Folge haben. Um diesen Anforderungen gerecht zu werden, wurden zwei Schlüsselpaare eingeführt: Ein Zone-Signing-Key (ZSK), der den Inhalt einer Zone signiert, und ein Key-Signing-Key, der nur den öffentlichen Teil der Schlüssel, insbesondere des Zone-Signing-Keys, signiert und bei der Delegation verwendet wird. Da der Key-Signing-Key (KSK) nicht für alltäglich anfallende DNS-Änderungen, sondern nur bei einem Schlüsselwechsel benötigt wird, ist es leichter, ihn auf einem Rechner ohne Netzzugang einzusetzen und die Signaturen der Keys mit einem Datenträger auf den autoritativen Server zu übertragen; somit wird ein besserer Schutz des privaten Key-Signing-Keys möglich. Diese Arbeitsweise wäre für den Zone-Signing-Key kaum denkbar; wie man allerdings im Abschnitt über den praktischen Betrieb am LRZ sehen wird, ist es auch möglich, die Zone-Signing-Keys nicht auf den weltweit erreichbaren autorativen Servern zu halten.

Um den Key-Signing-Key mit dem Vertrauensanker zu verknüpfen, wird ein Hash des Keys in der delegierenden Zone in Form eines DS-Records (Delegation Signer) gespeichert. Dieser Hash wird von der delegierenden Zone signiert und der entsprechende RRSIG-Record angelegt und damit die Signaturkette aufgebaut. Die beschriebene DNSSEC-PKI unterscheidet sich grundlegend von anderen PKIs, insbesondere von den Zertifizierungsstellen (Certificate Authorities bzw. CAs), die TLS-Zertifikate ausstellen. Anders als bei den TLS-CAs, die aus technischer Sicht keine Einschränkungen bezüglich ihrer Zertifizierungsobjekte haben und denen man damit ein sehr weitreichendes Vertrauen entgegenbringen muss, ist bei DNSSEC der Anwendungsbereich durch den DNS-Baum festgelegt: Nur den Verwaltern der Root-Zonen-Schlüssel muss aus technischer Sicht vollständig vertraut werden, die Hoheit über die darunter liegenden Domains ist auf die Betreiber der jeweiligen Zonen beschränkt. Eine de-Domain kann also nur über den Verwalter der de-Zone (DENIC) sicher delegiert werden, während man ein TLS-Zertifikat für DE-Domains bei vielen (nicht auf Deutschland beschränkten) CAs bekommt. Ein weiterer grundlegender Unterschied ist der Umgang mit Schlüsseln, die nicht mehr vertrauenswürdig sind. Während bei TLS jedes nicht mehr vertrauenswürdige Zertifikat zurückgerufen bzw. gesperrt werden muss und ständig wachsende Sperrlisten (Certificate Revocation Lists) entstehen, ist ein Rückruf bei DNS in dieser Form nicht möglich. Um Schlüssel ungültig zu machen, müssen sie ausgetauscht, und die dazugehörenden DNS-Records (DNSKEY und besonders DS, als Folge auch die RRSIGs) aktualisiert werden. Je nach Angriff bekommt das Opfer von diesem Austausch zunächst aber gar nichts mit, auch der Ablauf der TTL des DS-Records (der in einer nicht kompromittierten Zone liegt) hilft nicht immer weiter, da der Angreifer den alten DS-Record mit der alten Signatur wiederverwenden kann, solange die Signatur noch gültig ist. Das Fehlen von Sperrlisten muss daher durch eine passende Gültigkeitsdauer der Signaturen und, je nach Sicherheitsbedarf, durch regelmäßiges Austauschen der Zone-Signing-Keys berücksichtigt werden. Da die Gültigkeit der Signaturen auch von der Zeit (Zeitpunkt des Signierens, aktuelle Zeit auf dem validierenden Resolver) abhängt, muss auch die Uhr der DNS-Server überwacht und vor Drift bzw. Manipulationen geschützt werden.

Wie der DNS-Resolver die Authentizität eines Records sicherstellt, soll an dem A-Record für wlan.lrz.de verdeutlicht werden (die Zeit wurde im Beispiel nicht berücksichtigt, vgl. Abbildung 1):

1. Der Resolver fragt die Root-Zone nach dem DNSKEY (hier dem Zone-Signing-Key) und dem dazu gehörenden RRSIG-Record.

1a. Die RRSIGs aus Schritt 1 werden anhand des lokal gespeicherten öffentlichen Key-Signing-Key des Root-Servers verifiziert.

2. Der Resolver fragt in der Root-Zone nach den autoritativen DNS-Servern für die de-Zone (d. h. nach NS-Records, RRSIG-Records zu den NS-Records, DS-Record und RRSIG-Record zum DS-Record).

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

2a. Die RRSIGs aus Schritt 2 werden anhand des in Schritt 1a verifizierten DNSKEY verifiziert.
3. Einer der autoritativer DNS-Server für die de-Zone wird nach den DNSKEYs für die de-Zone, den NS-Records für die Zone lrz.de, dem DS-Record für lrz.de und jeweils nach den RRSIGs dazu gefragt.
3a. Der DNSKEY vom Typ Key-Signing-Key wird gehasht und mit dem bereits verifizierten DS-Record für die de-Zone aus der Root-Zone verglichen.
3b. Der DNSKEY vom Typ Zone-Signing-Key für die de-Zone wird anhand des in Schritt 3a verifizierten DNSKEY verifiziert.
3c. Die RRSIGs der NS-Records und des DS-Records für lrz.de werden anhand des DNSKEY aus Schritt 3b verifiziert.
4. Einer der autoritativen DNS-Server für lrz.de wird nach den DNSKEYs und dem A-Record zu wlan.lrz.de sowie den dazugehörigen RRSIGs gefragt.
4a. Der DNSKEY vom Typ Key-Signing-Key wird gehasht und mit dem in Schritt 3c verifizierten DS-Record verglichen.
4b. Der DNSKEY vom Typ Zone-Signing-Key wird anhand des Schlüssels aus Schritt 4a verifiziert.
4c. wlan.lrz.de wird anhand des Schlüssels aus Schritt 4b verifiziert

Damit ist eine Signaturkette vom Key-Signing-Key des Root-Servers, d. h. vom Vertrauensanker, zur Signatur des A-Records von wlan.lrz.de verifiziert.

Pro und contra DNSSEC

Wie oben beschrieben, liegt der wesentliche Vorteil von DNSSEC darin, dass die Authentizität der DNS-Daten garantiert werden kann. Bedenkt man die kaum zu überschätzende Bedeutung von DNS für das Internet und die unzähligen Angriffsmöglichkeiten auf andere Dienste über DNS, so ist das allein schon ausreichend, DNSSEC trotz der möglichen Schwierigkeiten einzusetzen. Hinzu kommen viele neue Anwendungsmöglichkeiten, die alle darauf hinauslaufen, dass man im DNS verfügbare Daten bestimmten Personen oder Organisationen durch kryptographisch sichere Authentifizierung zuordnen kann. Prominentere Beispiele sind TLS-Fingerprints (z. B. von Web- oder Mail-Servern), SSH-Fingerprints für SSH-Server oder öffentliche PGP-Schlüssel zu einer bestimmten E-Mail-Adresse, die man in unterschiedlicher Form (TLSA-Record, SSHFP-Record, bald auch OPENPGPKEY-Record) im DNS verfügbar machen und mit DNSSEC validieren kann (siehe Abschnitt „Mehrwert für andere Dienste“).

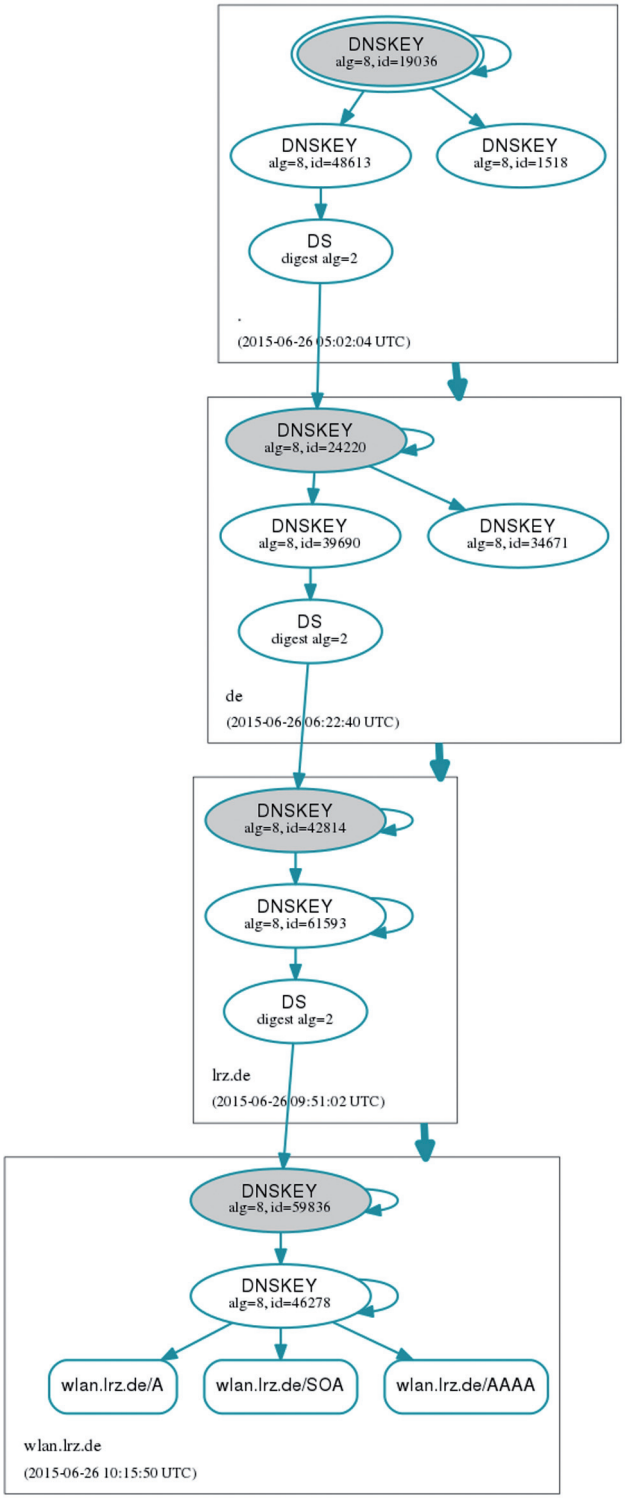

Abb. 1: DNSSEC-Signaturkette am Beispiel von wlan.lrz.de DNSViz.

Dem gegenüber stehen jedoch auch Kritikpunkte, die abgewogen werden müssen. Als Hauptproblem von DNSSEC wird die Komplexität gesehen. Die Abläufe und Abhängigkeiten im Protokoll sind umfangreich; selbst erfahrene Administratoren sind bei der Fehlersuche auf einen exter-

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

nen Blick auf die eigene Zone angewiesen, wie ihn z. B. Werkzeuge wie DNSViz bieten.

Wie bei den meisten Sicherheitsmechanismen kann das System keine Unterscheidung zwischen einer fehlerhaften Signatur und einem mutwilligen Angriff treffen und muss im Zweifelsfall von einem Angriff ausgehen. Insbesondere die Tatsache, dass RRSIG-Records eine vordefinierte Gültigkeitsdauer besitzen und vor deren Ablauf neu generiert werden müssen, hat schon einige Ausfälle, auch auf der Ebene der Top-Level-Domains, verursacht. Eine unvollständige Liste findet man auf IANIX. Auch der periodisch empfohlene Schlüsselwechsel (*Key-Rollover*) birgt immer wieder Gefahren.

DNS wird oft für UDP-Reflection-Attacks (in Form einer DNS-Amplification-Attack) missbraucht. Durch DNSSEC und die damit einhergehenden vergleichsweise großen Records (ca. 500 Bytes für eine Signatur) steigt nicht nur die Last der DNS-Server, auch ein Missbrauch der autoritativen Server für DNS-Amplification-Attacks wird erleichtert bzw. effektiver gemacht. Ganz abstellen kann man diese Art von Missbrauch nicht, jedenfalls nicht auf DNS-Ebene, allerdings kann man ihn durch Rate-Limiting einschränken: Aktuelle Versionen der DNS-Server-Software bind können die Query-Rate so einschränken, dass der reguläre Betrieb eines Resolvers nicht beeinträchtigt wird, Angriffe aber in ihrer Wirkung (zumindest bezogen auf diesen Server) eingeschränkt werden.

Weitere Kritik entzündet sich an der Tatsache, dass es nur einen einzigen Vertrauensanker in der Root-Zone gibt, der zudem unter der indirekten Kontrolle der US-Regierung steht. Dieser Kritikpunkt ist aus unserer Sicht schwach, da es derzeit keine skalierbare Alternative gibt, die nicht ebenfalls durch die US-Regierung kontrollierbar wäre. Gerade CAs haben (seit Snowden auch mehr oder weniger offiziell) im Hinblick auf staatliche Einmischung einen schlechten Track-Record. Abgesehen davon ist es technisch möglich, weitere Vertrauensanker auf einem Resolver zu konfigurieren, indem die öffentlichen Key-Signing-Keys einer Domain statisch zugeordnet werden (Konfigurationsoption „dnssec-lookaside“ für ISC-bind). Key-Infrastructure-Modelle, die frei vom Verdacht der Einmischung durch Staat oder Unternehmen sind, erfordern üblicherweise das individuelle Prüfen letztendlich jedes öffentlichen Schlüssels. Diesen Aufwand halten viele schon bei wenigen E-Mail-Schlüsseln für unzumutbar, für DNS wäre der Ansatz völlig unpraktikabel.

Auch die Betreiber der Top-Level-Domains (wie den Länderdomains) können in die Vertrauenskette eingreifen. Ein oft gewähltes Beispiel ist der beliebte Kurz-URL-Dienst bit.ly unter der lybischen Länderdomain, deren Betrieb durch die instabile Situation vor Ort beeinträchtigt werden könnte. Dabei wird jedoch gerne vergessen, dass der Betreiber einer Top-Level-Domain auch ohne DNSSEC in die Delegation eingreifen und Domains komplett übernehmen kann. Dass das auch anderenorts und ohne DNSSEC passieren kann, zeigt die Beschlagnahmung von No-IP-Domains durch Microsoft ms-no-ip, daher sollte dieses Argument bei der Diskussion von DNSSEC keine große Rolle spielen.

Kryptographen macht nicht zuletzt die sehr konservative Wahl der Schlüsselgrößen und Algorithmen in der Rootzone Sorgen (KSK 2048 Bit RSA/SHA-256, ZSK gar nur 1024 Bit RSA/SHA-256). RFC 6781 erlaubt die Nutzung von 1024 Bit RSA-Schlüsselpaaren derzeit noch explizit, an dieser Aussage regt sich jedoch Kritik und weckt gerade in der Rootzone Misstrauen. Unter anderem für die Entwickler von Google Chrome ist die Nutzung dieser Schlüssellänge derzeit ein Argument gegen DANE ImperialViolet.

Mehrwert für andere Dienste

Ein weit verbreiteter Standard für authentifizierten und verschlüsselten Datenaustausch ist derzeit SSL bzw. TLS RFC 5246 mit der dazugehörenden CA-Infrastruktur. Der Browser überprüft, beispielsweise beim Online-Banking, über das TLS-Zertifikat zwar den Domain-Namen, allerdings erst auf http-Ebene. Abgesehen davon, dass man sein Vertrauen sehr breit streut, indem man den „üblichen“ (d. h. den auf den meisten Betriebssystemen und in den Browsern vorinstallierten) CAs vertraut, bleibt die DNS-Auflösung von TLS unberührt bzw. ungesichert und ist damit ein Angriffsvektor, den man zusammen mit einem kompromittierten TLS-Zertifikat ausnutzen kann. Bei Webseiten funktioniert TLS trotzdem so gut, dass es sehr breit eingesetzt wird, daher profitieren von DNSSEC in erster Linie Dienste, in denen TLS und die CA-Infrastruktur nicht verfügbar oder weniger etabliert sind.

Eine flächendeckende Implementierung von DNSSEC bietet die Sicherheit, dass die angefragten DNS-Inhalte nur durch die Administratoren der beteiligten DNS-Zonen verändert werden können. Die kryptographische Absicherung ermöglicht die Veröffentlichung von Informationen im DNS, die den Eigentümern der Domain zugeschrieben werden kann. Den Eigentümern muss man in der Regel (z. B. im Falle von Mail) ohnehin in gewissem Umfang vertrauen, dieses Vertrauen wird durch DNSSEC nutzbar gemacht, wie folgende Beispiele zeigen sollen:

1. SSH-Server-Fingerprint/SSFP-Record RFC 4255
2. OpenPGP-Schlüssel (OPENPGPKEY-Record) DANE-OPENPGPKEY
3. TLS-Fingerprint (TLSA-Record) RFC 6698

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

Der Hauptanwendungszweck dieser neuen DNS-Records liegt in der Unterstützung von asymmetrischen Verschlüsselungsverfahren wie RSA. Diese sind nur dann sicher, wenn der in der Kommunikation verwendete öffentliche Schlüssel tatsächlich zweifelsfrei dem gewünschten Kommunikationspartner zugeordnet werden kann. Zu diesem Zweck gibt es je nach Protokoll verschiedene Ansätze: Die bereits genannte Hierarchie von vertrauenswürdigen Zertifizierungsstellen in einer TLS-PKI, eine gegenseitige Vertrauensbezeugung (Web of Trust) bei PGP oder die manuelle Prüfung von langen Fingerprints bei SSH.

DNS bietet im Zusammenhang mit DNSSEC die Möglichkeit, diese Informationen zu veröffentlichen und weltweit zugreifbar zu machen. SSH profitiert durch die Validierung von SSH-Hostkeys über SSHFP-Records. Bei SSH sind üblicherweise keine Zertifizierungsstellen im Einsatz (auch wenn es vom Protokoll unterstützt wird). Der Benutzer bekommt bei der ersten Verbindung mit einem Host einen kryptographischen Hash des öffentlichen Schlüssels präsentiert und sollte diesen Fingerprint mit dem erwarteten Schlüssel abgleichen. Aus Bequemlichkeit oder Unkenntnis vergleichen die meisten Nutzer den Fingerprint nur oberflächlich oder gar nicht. Durch die Hinterlegung des erwarteten Fingerprints im DNS (und dessen Validierung auf Authentizität durch DNSSEC) kann dieser manuelle und fehleranfällige Prozess von SSH-Clients (Option „VerifyHostKeyDNS“ bei OpenSSH) zuverlässig automatisiert werden. Dies ist besonders interessant für SSH-Zugänge, die von vielen verschiedenen, unter Umständen auch organisationsfremden Nutzern verwendet werden. Am LRZ sind SSHFP-Records beispielsweise für die Login-Knoten des europaweit genutzten Höchstleistungsrechners SuperMUC in Verwendung, aber auch bei der Nutzung von Administrationszugängen auf eigenen Servern ist die automatische Überprüfung sehr praktisch. Für OpenPGP wird gerade der OPENPGPKEY-Record eingeführt openpgpkey-draft, der einer E-Mail-Adresse (über DNS und mit DNSSEC gesichert) einen öffentlichen PGP-Schlüssel zuordnet und somit nicht nur den aufwändigen (und oft nicht durchgeführten) Vergleich der Schlüssel auf sicherem Wege ersetzen kann, sondern vollständig automatisierte Ende-zu-Ende-Verschlüsselung für Mails ermöglicht, indem der Mail-Client prinzipiell ohne Nutzereingriffe den richtigen PGP-Schlüssel erfragen und verwenden kann. Für die Verifikation von SSL/TLS-Schlüsseln existiert mit DANE (DNS-Based Authentication of Named Entities) [RFC6698] [RFC6698] die Möglichkeit, den Public-Key zu jedem Dienst, d. h. zu jeder (Host, Protokoll, Port)-Kombination, im DNS zu hinterlegen und beim Verbindungsaufbau zu validieren. Diese Methode wird insbesondere in Deutschland schon von vielen Providern eingesetzt, um die E-Mail Kommunikation abzusichern und damit gegen Abhören bzw. Man-in-the-Middle Angriffe zu schützen. Auch staatliche Stellen wie die Bundesregierung (bund.de) oder der Freistaat Bayern (bayern.de) haben diese Möglichkeit bereits implementiert. Auf DANE und Verschlüsselung des Mailtransports wird in einem späteren Artikel gesondert eingegangen.

Betriebsaspekte von DNSSEC am Beispiel des LRZs

Das LRZ setzt DNSSEC mit validierenden Resolvern seit 2008 im Münchner Wissenschaftsnetz (MWN) ein. Nach der Teilnahme an DNSSEC-Testbeds und dem erfolgreichem Signieren kleinerer Domains seit 2010 hat das Thema im letzten Jahr massiv an Bedeutung gewonnen. Seit März 2014 werden alle LRZ-Resolver DNSSEC-validiert betrieben und innerhalb von drei Monaten wurden bis Januar 2015 auch die großen Hauptdomains lrz.de, tum.de (Technische Universität München) und lmu.de (Ludwig-Maximilians-Universität München) signiert. Im folgenden Abschnitt werden das Setup im MWN sowie Betriebs-, Monitoring- und Managementaspekte vorgestellt.

Das LRZ betreibt für seine Kunden eine DNS-Server- und -Resolver-Infrastruktur, mit der über 1.500 Domains und 12.000 Subdomains verwaltet werden. Die knapp 350 Kunden können ihre Domains mandantenfähig per Web-Frontend auf Basis des Produkts Nixu (jetzt FusionLayer, Inc.) NameSurfer selbst verwalten. Die Server werden redundant und ausfallsicher an verschiedenen Kernnetzstandorten innerhalb des MWN und auf einem Server in Portugal betrieben (vgl. Abbildung 2). Auf den physischen Servern laufen autoritative DNS-Server und der Resolver-Dienst jeweils auf einer eigenen Anycast-Adresse. Hinter dns1.lrz.de, dns2.lrz.bayern, resolver1.lrz.de und resolver2.lrz.de verbergen sich je zwei Server, die über Anycast erreichbar sind und sich die Last teilen. Fällt ein Server aus, so verfällt automatisch die Anycast-Route dieses Servers und die IP-Adresse wird überall zum anderen Server geroutet. Damit sich Probleme in einer Domain (z. B. lrz.de) nicht über die NS-Records auf alle anderen Domains ausweiten, wurden die Nameserver in verschiedene Top-Level-Domains eingetragen. Zum Beispiel zeigen für die Domäne lrz.de die Nameserver-Einträge auf dns1.lrz.de, dns2.lrz.bayern und dns3.lrz.eu. Als Software kommen ISC BIND, Nixu Namesurfer und verschiedene Eigenentwicklungen für Administration und Monitoring zum Einsatz.

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

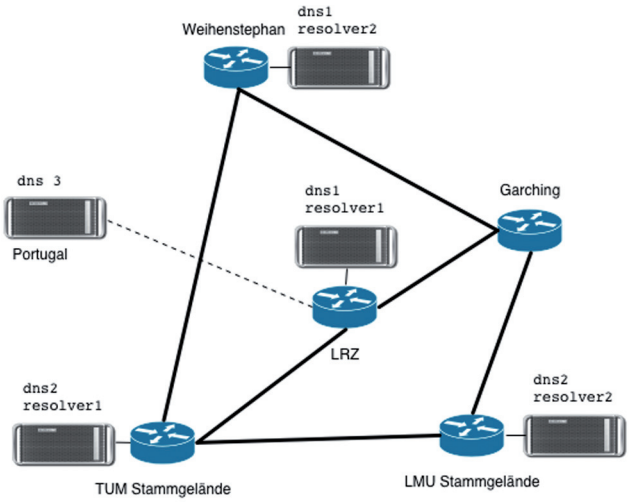

Abb. 2: Die DNS-Server-Infrastruktur im Münchner Wissenschaftsnetz.

Die Umstellung auf DNSSEC-Validierung birgt das Risiko, dass Einträge aus fehlerhaft DNSSEC-signierten Zonen gar nicht mehr aufgelöst werden können und diese Zonen damit für alle Clients validierender Resolver nicht mehr erreichbar sind. Fehler in Signaturen und Signaturketten traten, bedingt durch die höhere Komplexität und nicht ausgereifte Software, zu Beginn noch häufiger auf und fielen aus Nutzersicht, oft zu Unrecht, auf die Betreiber der validierenden Resolver zurück. Daher hatte das LRZ zunächst nur einen seiner Resolver entsprechend konfiguriert, um die Systemreife beurteilen zu können, ohne die Dienstnutzung durch angebundene Endgeräte zu gefährden.

Im April 2014 wurden dann alle Resolver auf DNSSEC-Validierung umgestellt und trotz der relativ großen Anzahl an versorgten Benutzern sind im Laufe eines Jahres nur fünf Störungsmeldungen eingegangen, die jedoch nur peripher durch DNSSEC hervorgerufen wurden und deren Ursache immer auf der jeweiligen Gegenseite lag. Trotz dieser Zwangsbeglückung aller Anwender mit DNSSEC sind also keine größeren Probleme aufgetreten.

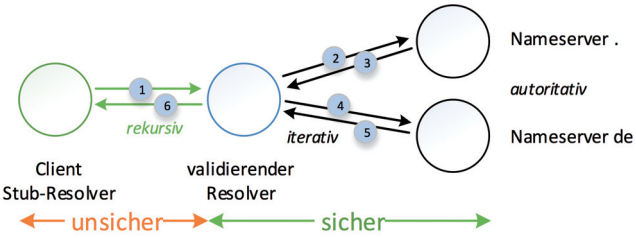

Abb. 3: Unsichere Client-Stub-Revolver.

Erwähnenswert bleiben zwei Schwierigkeiten: Zum einen können DNSSEC-validierende Resolver auch weiterhin kein DNS-Spoofing auf dem Weg zwischen Endgerät und DNS-Resolver verhindern (vgl. Abbildung 3). Das Problem ist dann gelöst, wenn man dem Netz zwischen Endgerät und Resolver vertraut, im Idealfall sollte sich der Resolver also auf dem jeweiligen Endgerät befinden. Im LRZ kommt deshalb bereits auf einigen Produktivsystemen als lokaler validierender Resolver die OpenSource-Software Unbound zum Einsatz (vgl. Abbildung 4). Zum anderen werden lokale DNS-Zonen, für die der Resolver als Slave fungiert, von ISC-BIND nicht DNSSEC-validiert. Um dieses Problem zu umgehen, müssten die Resolver ihren Slave-Status verlieren; das ist bei den LRZ-Resolvern noch nicht der Fall, da DNS-Änderungen damit nicht sofort, sondern erst nach Ablauf der TTL wirksam würden.

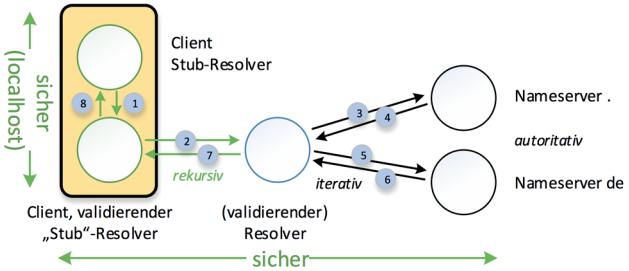

Abb. 4: Einsatz validierender Stub-Resolver.

Auf der autoritativen Seite gibt es derzeit zwei weitverbreitete freie Ansätze: die Software OpenDNSSEC und die ab ISC BIND 9.8 integrierte automatische Signierung. Beide Produkte können die Zone bei Änderungen oder auslaufenden Signaturen automatisch neu signieren, wodurch abgelaufene oder fehlende Signaturen (eine „beliebte" Fehlerquelle) vermieden werden. Während OpenDNSSEC schon immer hauptsächlich als Signing-Proxy (d. h. als Zwischensystem, welches die unsignierte Zone von einem anderen Server abholt, Signaturen und Schlüssel hinzufügt und die signierte Zone für andere Server zur Verfügung stellt) arbeitete, ist diese Funktionsweise (*inline-signing*) für ISC Bind erst ab Version 9.9 verfügbar. OpenDNSSEC bietet zwar noch mehr Funktionalität im Bereich der Schlüsselverwaltung, hat aber Abhängigkeiten zu einer externen Datenbank und benötigt ein Hardware-Security-Modul (*HSM*), welches paradoxerweise auch in Software implementiert werden kann. Aufgrund der höheren Komplexität wurde daher am LRZ auf BIND 9.9 als Signing-Proxy gesetzt, welcher nur für DNSSEC-aktivierte Zonen zwischen der vom Kunden nutzbaren Weboberfläche und den autoritativen Servern steht (vgl. Abbildung 5). Das Schlüsselmaterial liegt dabei nur auf dem

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

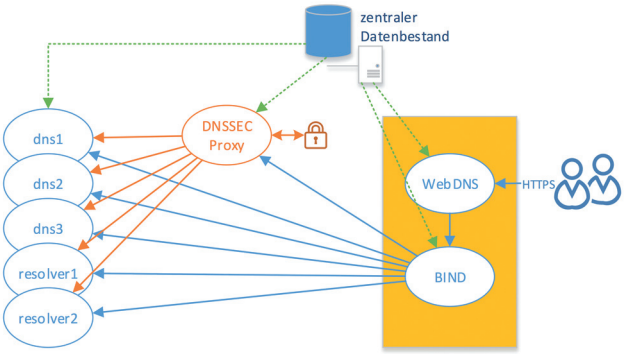

Abb. 5: Einsatz des DNSSEC-Proxy am LRZ.

Signing-Proxy, der in einem abgeschotteten Netzsegment steht und gegen Angriffe gehärtet ist. Als Algorithmus wurde der konservative Ansatz RSA/SHA256 mit 2048 Bit Schlüssellänge gewählt, da dieser die weiteste Verbreitung hat. EC-basierte Algorithmen haben zwar Vorteile bei der Länge der Signatur und der Rechenleistung, sind aber auf der Validierungsseite noch nicht so weit verbreitet.

Bei Problemen oder Fehlern im Betrieb von DNSSEC kann es dazu kommen, dass im schlimmsten Fall die gesamte Domäne von validierenden Resolvern nicht mehr aufgelöst wird und damit für die Nutzer die Domäne überhaupt nicht mehr erreichbar ist. Die Weltkart in Abbildung (Abbildung 6) zeigt, wie viel Prozent der getesteten Nutzer im jeweiligen Land DNSSEC nutzen. Diese Karte basiert auf Arbeiten von Geoff Huston der ein Konzept zur Messung von DNS aus Nutzerpersepektive entwickelt hat Huston 2014.Einer der wichtigsten Aspekte von DNSSEC ist jedoch das Monitoring. Es zeigen sich bei DNSSEC ähnliche Probleme wie in den Anfängen von IPv6; da es anteilsmäßig nur wenige Nutzer gibt, ist die Zone bei Fehlern nicht vollständig offline, sondern nur von wenigen Nutzern nicht mehr auflösbar. Dadurch suchen die Betreiber den Fehler erst einmal auf Nutzerseite und empfehlen im Zweifelsfall eine Deaktivierung der Validierung. Tatsächlich bemerkt der Betreiber den Fehler unter Umständen gar nicht selbst, wenn er wie das LRZ seine eigenen Zonen als Kopie auf den Resolvern hält.

Um dieses Problem zu verhindern wurde bei der Einführung von DNSSEC auch ein intensives Monitoring implementiert. Einer der wichtigsten Tests ist dabei die regelmäßige Abfrage der eigenen DNS-Daten bei externen DNSSEC-validierenden Resolvern. Hier bieten sich neben dem bekannten Dienst Google DNS auch die öffentlich erreichbaren Referenzserver im DNS-OARC ODVR-Projekt an. Fehler in den eigenen Zonen resultieren hier in Abfra-

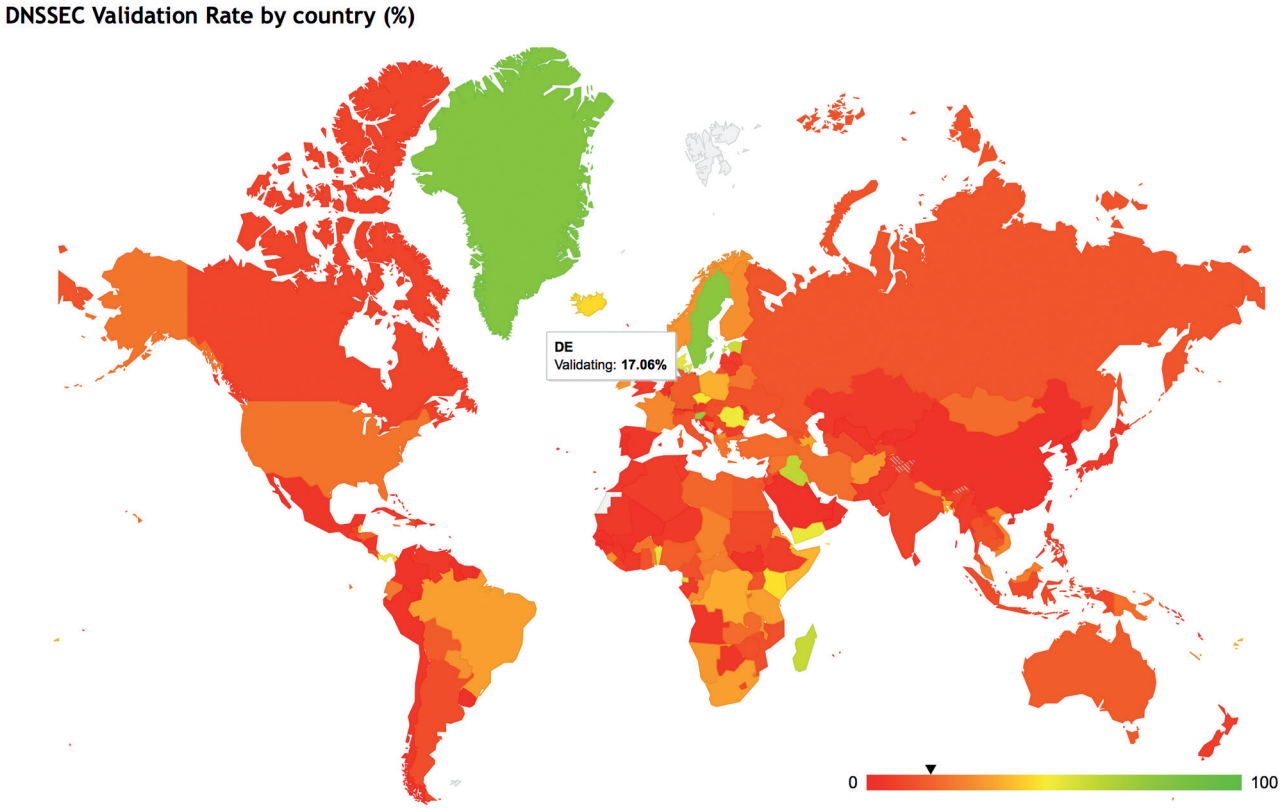

Abb. 6: Validierungsrate von DNSSEC pro Land DNSSEC Rate.

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

gefehlern, die sofort an die Administratoren gemeldet werden können. Nutzt man lokale validierende Resolver, tauchen Fehler auch hier auf, was das klassische Monitoring bemerken sollte. Zusätzlich werden die vollständigen signierten Zonendaten täglich durch die Prüftools von LDNS (*ldns-verify-zone*) und BIND (*dnssec-verify*) auf Konsistenz geprüft. Hierbei werden auch Signaturen, die in weniger als zwei Wochen ihre Gültigkeit verlieren, gemeldet. Da in der LRZ-Konfiguration Signaturen drei Wochen vor Ablauf erneuert werden, wäre diese Warnung ein starker Hinweis auf einen grundlegenden Fehler im Gesamtaufbau.

Fazit und Handlungsempfehlung

DNSSEC löst nicht alle Designprobleme, die seit der Entstehungszeit des DNS-Protokolls identifiziert wurden. So fehlt beispielsweise weiterhin vertrauliche Kommunikation auf Basis von Verschlüsselung, was allerdings einerseits im Interesse der Performance liegt und andererseits in letzter Konsequenz einen grundlegend anderen global verteilten Architekturansatz benötigen würde, wie er beispielsweise vom GNU Name System GNS verfolgt wird. Die von DNSSEC ermöglichte signaturbasierte Integritäts- und Authentizitätssicherung löst aber mit DNS-Spoofing das durch viele bekannt gewordene Sicherheitsvorfälle dringendste Problem von DNS, das wie ein Damoklesschwert über allen herkömmlichen DNS-Servern hängt.

Obwohl Konzepte und Implementierungen inzwischen ausgereift und praxistauglich sind, steigt die Komplexität signifikant: Kleine Fehler bei der Administration führen schnell zu ungültigen Signaturen und somit faktisch zu einer Nichterreichbarkeit der eigenen Dienste. Im Real-World-Betrieb sind deshalb zuverlässige DNSSEC-Werkzeuge, die Managementworkflows weitestgehend automatisieren und Prüfungen durchführen, genauso wichtig wie ein Monitoring der eigenen DNS-Zonen von innen und insbesondere auch von außen. Der Lohn für die nicht zu unterschätzende Lernkurve und den oftmals holprigen Start bei der Umstellung der ersten eigenen Zonen auf DNSSEC ist jedoch nicht nur die abgesicherte Basisfunktionalität bei der DNS-basierten Namensauflösung; vielmehr ergeben sich mit dem Hinterlegen von User- und Serverzertifikaten sowie Fingerprints interessante neue Möglichkeiten, von denen andere Protokolle wie SSH und globale Infrastrukturen z. B. für den E-Mailversand sowohl sicherheitstechnisch als auch bezüglich Benutzerfreundlichkeit massiv profitieren können.

Die Ausschöpfung dieses Potentials ist maßgeblich von der Durchdringungsrate von DNSSEC abhängig; mit unter 15% des weltweiten DNS-Verkehrs ist der Durchbruch noch lange nicht erzielt. Dennoch ist ähnlich zu anderen modernisierten Protokollvarianten wie HTTP/2 und IPv6 davon auszugehen, dass mittelfristig kein Weg daran vorbei führen wird. Das Beispiel des Münchner Wissenschaftsnetzes zeigt, dass ein stabiler DNSSEC-Betrieb auch in größeren Umgebungen bereits problemlos möglich ist und praktischen Mehrwert bietet. Eine frühzeitige Auseinandersetzung mit der Thematik kann deshalb ebenso wie das Sammeln eigener Erfahrungswerte uneingeschränkt empfohlen werden.

Literatur

DANE-OPENPGPKEY P. Wouters: IETF-Draft – Using DANE to Associate OpenPGP public keys with email addresses draft-ietf-dane-openpgpkey-03, 2015. https://tools.ietf.org/html/draft-ietf-dane-openpgpkey-03

dns-axfr Internetwache: Scanning Alexa's Top 1M for AXFR, 2015. https://en.internetwache.org/scanning-alexas-top-1m-for-axfr-29-03-2015/

DNSKam Friedl, Steve. An Illustrated Guide to the Kaminsky DNS Vulnerability, August 2008. http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

DNS-OARC DNS-OARC Domain Name System Operations Analysis and Research Center. https://www.dns-oarc.net/oarc/services/odvr

DNSSEC Rate APNIC: DNSSEC Validation Rate by country. http://gronggrong.rand.apnic.net/cgi-bin/worldmap

Dnsviz DNSViz. http://dnsviz.net

GNS GNU's Framework for Secure Peer-to-Peer Networking: The GNU Name System, 2012. https://www.gnunet.org/gns

Huston 2014 Geoff Huston: Measuring the DNS from the Users' perspective. RIPE 68, May 2015, Warsaw, Poland. https://ripe68.ripe.net/presentations/164-2014-05-14-huston-dns-measurements.pdf

iana-dnssec-alg IANA: Domain Name System Security (DNSSEC) Algorithm Numbers, 2014. http://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xhtml

IANIX IANIX: List of Major DNSSEC Outages and Validation Failures, 2015. http://ianix.com/pub/dnssec-outages.html

ImperialViolet ImperialViolet: Why not DANE in browsers, 2015. https://www.imperialviolet.org/2015/01/17/notdane.html

ISC-BIND Internet Systems Consortium (ISC): BIND The most widely used Name Server Software. https://www.isc.org/downloads/bind/

LDNS NLnet Labs: LDNS. http://www.nlnetlabs.nl/projects/ldns/

ms-no-ip Heise Newsticker: Malware: Microsoft erzwingt Umleitung von Domains des DynDNS-Diensts

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21

NoIP, 2014. http://www.heise.de/newsticker/meldung/Malware-Microsoft-erzwingt-Umleitung-von-Domains-des-DynDNS-Diensts-NoIP-2243605.html

openpgpkey-draft P. Wouters: Internet-Draft DANE-OpenPGP: Using DANE to Associate OpenPGP public keys with email addresses, 2015. https://tools.ietf.org/html/draft-ietf-dane-openpgpkey-02

RFC 920 J. Postel, J. Reynolds: RFC 920 – Domain Requirements, 1984. https://tools.ietf.org/html/rfc920

RFC 4033 R. Arends, R. Austein, M. Larson, D. Massex, S. Rose: RFC 4033 – DNS Security Introduction and Requirements, 2005. http://tools.ietf.org/html/rfc4033

RFC 4034 R. Arends, R. Austein, M. Larson, D. Massex, S. Rose: RFC 4034 – Resource Records for the DNS Security Extensions, 2005. http://tools.ietf.org/html/rfc4034

RFC 4035 R. Arends, R. Austein, M. Larson, D. Massex, S. Rose: RFC 4035 – Protocol Modifications for the DNS Security Extensions, 2005. http://tools.ietf.org/html/rfc4035

RFC 4255 Schlyter, W. Griffin: RFC 4255 – Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints, 2006. http://tools.ietf.org/html/rfc4255

RFC 5246 T. Dierks, E. Rescorla: RFC 5246 – The Transport Layer Security (TLS) Protocol Version 1.2, 2008. http://tools.ietf.org/html/rfc5246

RFC 5280 D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk: RFC 5280 – Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, 2008. https://tools.ietf.org/html/rfc5280

RFC 5936 E. Lewis, A. Hoenes: RFC 5936 – DNS Zone Transfer Protocol (AXFR), 2010. https://tools.ietf.org/html/rfc5936

RFC 6698 Hoffman, P. & Schlyter, J.: The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA, August 2012. https://tools.ietf.org/html/rfc6698

Unbound Unbound. https://www.unbound.net/index.html

Helmut Reiser: Leibniz Supercomputing Centre – Networking, Garching n. Munich, Bavaria, Germany

Daniel Feuchtinger: Leibniz Supercomputing Centre – Networking, Garching n. Munich, Bavaria, Germany

Wolfgang Hommel: Leibniz Supercomputing Centre, Boltzmannstraße 1, Garching n. Munich, Bayern 85748, Germany

Bernhard Schmidt: Leibniz Supercomputing Centre, Boltzmannstraße 1, Garching n. Munich, Bayern 85748, Germany

Michael Storz: Leibniz Supercomputing Centre – Networking, Garching n. Munich, Bavaria, Germany

Angemeldet | reiser@lrz.de Autorenexemplar
Heruntergeladen am | 06.10.15 13:21