DNSSEC Deployment: From End-Customer to Content

ION San Diego
December 11, 2012
Our Panel Today

Moderator: Dan York, Internet Society

Panelists:

• Jim Galvin, Afilias
• Rick Lamb, ICANN
• Cricket Liu, Infoblox
• Roland M. van Rijsijik-Deij, SURFnet
Internet Society Deploy360 Programme

Providing real-world deployment info for IPv6, DNSSEC and other Internet technologies:

- Case Studies
- Tutorials
- Videos
- Whitepapers
- News, information

English content, initially, but will be translated into other languages.

www.internetsociety.org/deploy360/
What Problem Is DNSSEC Trying To Solve?

DNSSEC = "DNS Security Extensions"

- Defined in RFCs 4033, 4034, 4035
- Operational Practices: RFC 4641

Ensures that the information entered into DNS by the domain name holder is the SAME information retrieved from DNS by an end user.

Let's walk through an example to explain…
A Normal DNS Interaction

Web Server

Web Browser

DNS Resolver

1. example.com?

2. 10.1.1.123

3. https://example.com/

4. web page

Resolver checks its local cache. If it has the answer, it sends it back.

example.com 10.1.1.123

If not…
A Normal DNS Interaction

1. The Web Browser sends a DNS query to the DNS Resolver for `example.com`.
2. The DNS Resolver asks the DNS Svr for the `.com` root zone.
3. The DNS Svr returns the NS records for `.com`.
4. The DNS Resolver asks the DNS Svr for `example.com`.
5. The DNS Svr returns the NS records for `example.com`.
DNS Works On Speed

• First result received by a DNS resolver is treated as the correct answer.

• Opportunity is there for an attacker to be the first one to get an answer to the DNS resolver, either by:
  • Getting to the correct point in the network to provide faster responses;
  • Blocking the responses from the legitimate servers (ex. executing a Denial of Service attack against the legitimate servers to slow their responses)
Attacking DNS

1. DNS Resolver requests DNS records for example.com.
2. DNS Svr for example.com returns the DNS records.
3. The DNS Svr returns the IP address 192.168.2.2.
4. The Web Browser resolves the DNS records and requests the web page from 192.168.2.2.
5. The Web Server returns the web page.
6. The Web Browser displays the web page.

Key:
- DNS Svr: Domain Name Server
- .com: Top-level domain
- NS: Name Server
- root: Root DNS server
A Poisoned Cache

Web Server

Web Browser

DNS Resolver

Resolver cache now has wrong data:
example.com 192.168.2.2

This stays in the cache until the Time-To-Live (TTL) expires!
How Does DNSSEC Help?

• DNSSEC introduces new DNS records for a domain:
  • RRSIG – a signature ("hash") of a set of DNS records
  • DNSKEY – a public key that a resolver can use to validate RRSIG

• A DNSSEC-validating DNS resolver:
  • Uses DNSKEY to perform a hash calculation on received DNS records
  • Compares result with RRSIG records. If results match, records are the same as those transmitted. If the results do NOT match, they were potentially changed during the travel from the DNS server.
A DNSSEC Interaction

Web Server

https://example.com/

Web Browser

example.com?

DNS Resolver

10.1.1.123

DNS Svr root

DNS Svr .com

DNS Svr example.com

10.1.1.123

DNSKEY RRSIGs
But Can DNSSEC Be Spoofed?

• But why can't an attacker simply insert DNSKEY and RRSIG records? What prevents DNSSEC from being spoofed?

• An additional was introduced, the "Delegation Signer (DS)" record

• It is a fingerprint of the DNSKEY record that is sent to the TLD registry

• Provides a global "chain of trust" from the root of DNS down to the domain

• Attackers would have to compromise the registry
A DNSSEC Interaction

1. Web Browser requests example.com?
2. DNS Resolver queries .com root DNS SRV
3. .com DNS SRV returns example.com DNSKEY
4. DNS Resolver requests 10.1.1.123 example.com DNSKEY
5. Web Server provides https://example.com/
6. Web Browser displays web page
The Global Chain of Trust

1. DNS Resolver
2. DNS Svr .com
3. 10.1.1.123
4. Web Browser
5. Web Server
6. web page

https://example.com/
Attempting to Spoof DNS

Web Server

DNS Resolver

Web Browser

DNS Svr root

DNS Svr .com

DNS Svr example.com

DNS Svr .com

Attacking DNS Svr example.com

DNSKEY RR SIGs

example.com

example.com

DNSKEY RR SIGs

10.1.1.123

192.168.2.2

https://example.com/

example.com?

web page
Attempting to Spoof DNS

DNS Resolver

1. Example.com?
2. 0.1.1.123 DNSKEY RRSIGs
3. 192.168.2.2 DNSKEY RRSIGs
4. SERVFAIL
5. https://example.com/
6. Web page

Web Browser

Web Server

DNS Svr root

DNS Svr .com

DNS Svr example.com

Attacking DNS Svr example.com

example.com

.com NS DS

NS DS

example.com .com

DNS Svr example.com
What DNSSEC Proves:

• "These ARE the IP addresses you are looking for." (or they are not)

• Ensures that information entered into DNS by the domain name holder (or the operator of the DNS hosting service for the domain) is the **SAME** information that is received by the end user.
The Two Parts of DNSSEC

- Signing
  - Registries
  - Registrars
  - DNS Hosting

- Validating
  - Applications
  - Enterprises
  - ISPs
DNSSEC Signing - The Individual Steps

- **Registry**
  - Signs TLD
  - Accepts DS records
  - Publishes/signs records

- **Registrar**
  - Accepts DS records
  - Sends DS to registry
  - Provides UI for mgmt

- **DNS Hosting Provider**
  - Signs zones
  - Publishes all records
  - Provides UI for mgmt

- **Domain Name Registrant**
  - Enables DNSSEC (unless automatic)
Our Panel Today

Moderator: Dan York, Internet Society

Panelists:

• Jim Galvin, Afilias
• Rick Lamb, ICANN
• Cricket Liu, Infoblox
• Roland M. van Rijswijk-Deij, SURFnet
DNSSEC and SSL
Why Do I Need DNSSEC If I Have SSL?

• A common question: why do I need DNSSEC if I already have a SSL certificate? (or an "EV-SSL" certificate?)

• SSL (more formerly known today as Transport Layer Security (TLS)) solves a different issue – it provides encryption and protection of the communication between the browser and the web server.
The Typical TLS (SSL) Web Interaction

Web Server

Web Browser

DNS Resolver

DNS Svr root

DNS Svr .com

DNS Svr example.com

1. Web Browser requests DNS for example.com?
2. DNS Svr .com returns example.com
3. DNS Svr example.com returns 10.1.1.123
4. Web Browser connects to 10.1.1.123
5. Web Server returns https://example.com/
6. Web Browser receives TLS-encrypted web page
The Typical TLS (SSL) Web Interaction

1. Web Browser
2. DNS Resolver
3. DNS Svr example.com
4. 10.1.1.123
5. https://example.com/
6. TLS-encrypted web page

Is this encrypted with the CORRECT certificate?
What About This?

Web Server

DNS Server

Firewall
(or attacker)

Web Browser

TLS-encrypted web page with CORRECT certificate

https://www.example.com/

www.example.com?

1.2.3.4

1

2

TLS-encrypted web page with NEW certificate (re-signed by firewall)

www.internetsociety.org/deploy360/
Problems?

Web Server

DNS Server

Web Browser

Firewall

https://www.example.com/

TLS-encrypted web page with CORRECT certificate

www.example.com?

1.2.3.4

1

2

TLS-encrypted web page with NEW certificate (re-signed by firewall)

www.internetsociety.org/deploy360/
Problems?

Web Server

DNS Server

Firewall

Web Browser

https://www.example.com/

1.2.3.4

1

2

Log files or other servers

Potentially including personal information

TLS-encrypted web page with CORRECT certificate

TLS-encrypted web page with NEW certificate (re-signed by firewall)

www.example.com?
Issues

A Certificate Authority (CA) can sign ANY domain.

Now over 1,500 CAs – there have been compromises where valid certs were issued for domains.

Middle-boxes such as firewalls can re-sign sessions.
A Powerful Combination

- TLS = encryption + *limited* integrity protection
- DNSSEC = strong integrity protection

- How to get encryption + strong integrity protection?

- TLS + DNSSEC = **DANE**
DNS-Based Authentication of Named Entities (DANE)

• Q: How do you know if the TLS (SSL) certificate is the correct one the site wants you to use?

• A: Store the certificate (or fingerprint) in DNS (new TLSA record) and sign them with DNSSEC.

A browser that understand DNSSEC and DANE will then know when the required certificate is NOT being used.

Certificate stored in DNS is controlled by the domain name holder. It could be a certificate signed by a CA – or a self-signed certificate.
DANE

Web Server

DNS Server

Firewall (or attacker)

Log files or other servers

TLS-encrypted web page with CORRECT certificate

https://example.com/

TLS-encrypted web page with NEW certificate (re-signed by firewall)

DANE-equipped browser compares TLS certificate with what DNS / DNSSEC says it should be.

Web Browser w/DANE

example.com?

10.1.1.123

DNSKEY RRSIGs TLSA

www.internetsociety.org/deploy360/
DANE – Not Just For The Web

• DANE defines protocol for storing TLS certificates in DNS

• Securing Web transactions is the obvious use case

• Other uses also possible:
  • Email via S/MIME
  • VoIP
  • Jabber/XMPP
  • ?
DANE Resources

DANE Overview and Resources:

• http://www.internetsociety.org/deploy360/resources/dane/

IETF Journal article explaining DANE:


RFC 6394 - DANE Use Cases:

• http://tools.ietf.org/html/rfc6394

RFC 6698 – DANE Protocol:

• http://tools.ietf.org/html/rfc6698
How Do We Get DANE Deployed?

Developers:
- Add DANE support into applications (see list of libraries)

DNS Hosting Providers:
- Provide a way that customers can enter a “TLSA” record into DNS as defined in RFC 6698 (http://tools.ietf.org/html/rfc6698)
- This will start getting TLS certificates into DNS so that when browsers support DANE they will be able to do so.
- [More tools are needed to help create TLSA records – ex. hashslinger]

Network Operators / Enterprises / Governments:
- Start talking about need for DANE
- Express desire for DANE to app vendors (especially browsers)
Opportunities

• DANE is just one example of new opportunities brought about by DNSSEC

• Developers and others already exploring new ideas
Getting DNSSEC Deployed
Three Steps TLD Operators Can Take:

1. Sign your TLD!
   - Tools and services available to help automate process

2. Accept DS records
   - Make it as easy as possible (and accept multiple records)

3. Work with your registrars
   - Help them make it easy for DNS hosting providers and registrants

4. Help With Statistics
   - Can you help by providing statistics?

Implement DNSSEC and make your TLD more secure!
Three Requests For Network Operators

1. Deploy DNSSEC-validating DNS resolvers

2. Sign your own domains where possible

3. Help promote support of DANE protocol
   - Allow usage of TLSA record. Let browser vendors and others know you want to use DANE. Help raise awareness of how DANE and DNSSEC can make the Internet more secure.
Internet Society Deploy360 Programme

Can You Help Us With:

• Case Studies?
• Tutorials?
• Videos?

How Can We Help You?

www.internetsociety.org/deploy360/
Thank You!
Download A DNSSEC Whitepaper

“Challenges and Opportunities in Deploying DNSSEC”