Deploying DNSSEC: From End-Customer To Content

March 28, 2013
Our Panel

Moderator:

• Dan York, Senior Content Strategist, Internet Society

Panelists:

• Sanjeev Gupta, Principal Technical Architect, DCS1 Pte
• Jitender Kumar, Technical Account Manager, Afilias
• Richard Lamb, DNSSEC Program Manager, ICANN
A Quick Introduction to DNS and DNSSEC
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

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. DNS Resolver sends query to .com root server.
2. .com root server responds with NS record for example.com.
3. DNS Resolver sends query to example.com.
4. Example.com server returns IP address 10.1.1.123.
5. DNS Resolver returns IP address to Web Browser.
6. Web Browser fetches web page from IP address 10.1.1.123.
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 asks for example.com?
2. DNS Svr example.com answers with example.com NS to .com root.
3. DNS Svr .com receives request for example.com NS.
4. DNS Resolver receives 192.168.2.2 from example.com.
5. Web Browser requests https://example.com/.
6. Web Server returns the web page.

DNS Svr root
DNS Svr .com
DNS Svr example.com
Attacking DNS Svr example.com

10.1.1.123
192.168.2.2
A Poisoned Cache

Web Server

https://example.com/

Web Browser

example.com?

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.
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

Web Server

Web Browser

https://example.com/

DNS Resolver

example.com?

DNS Svr root

DNS Svr .com

DNS Svr example.com

10.1.1.123

10.1.1.123 DNSKEY RRSIGs

example.com NS

DS

.com NS

DS
The Global Chain of Trust

1. DNS Resolver
2. DNS Svr .com
3. 10.1.1.123
4. 10.1.1.123
5. https://example.com/
6. web page

Web Server
Web Browser
example.com

DNSKEY
RRSIGs

DNS Svr root
DNS Svr .com
DNS Svr example.com
Attempting to Spoof DNS

- Web Server
- Web Browser
- DNS Resolver
- DNS Svr example.com
- DNS Svr .com
- DNS Svr .com root

1. DNS Resolver requests example.com?
2. DNS Svr .com returns .com NS DS
3. DNS Svr .com returns example.com DNSKEY RRSIGs
4. DNS Svr example.com returns example.com NS DS
5. Web Browser requests https://example.com/
6. Web Server returns web page

Spoofing
- Attack DNS Svr 192.168.2.2 DNSKEY RRSIGs
- DNS Svr 10.1.1.123 DNSKEY RRSIGs
Attempting to Spoof DNS

1. DNS Resolver requests DNS records for example.com.
2. DNS Svr returns DNSKEY and RRSIGs for example.com.
3. DNS Svr returns DNSKEY and RRSIGs for .com.
4. DNS Svr returns SERVFAIL on further resolution.
5. Web Browser requests web page from example.com.

DNS Svr root
- .com
- NS
- DS

DNS Svr .com
- example.com
- NS
- DS

DNS Svr example.com
- 192.168.2.2
- DNSKEY
- RRSIGs

Attacking DNS Svr example.com
- 10.1.1.123
- DNSKEY
- RRSIGs
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
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

1. Web Browser sends request to DNS Resolver
2. DNS Resolver queries DNS Svr for example.com
3. DNS Svr returns 10.1.1.123
4. Web Browser connects to 10.1.1.123
5. Web Server returns TLS-encrypted web page
6. TLS-encrypted web page is displayed in the Web Browser
The Typical TLS (SSL) Web Interaction

1. DNS Resolver queries DNS Svr root for example.com.
2. DNS Svr root returns DNS Svr .com.
4. DNS Svr .com returns 10.1.1.123.
5. Web Browser requests https://example.com/.

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

Web Server

https://www.example.com/

Firewall (or attacker)

DNS Server

1.2.3.4

1

Web Browser

www.example.com?

2

TLS-encrypted web page with CORRECT certificate

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

www.internetsociety.org/deploy360/
Problems?

- Web Server
- DNS Server
- Firewall
- Web Browser

TLS-encrypted web page with CORRECT certificate

https://www.example.com/

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

www.example.com?

1.2.3.4

1

2
Problems?

Web Server \[\xrightarrow{\text{https://www.example.com/}}\]

DNS Server \[\xrightarrow{\text{www.example.com?}}\]

Firewall

https://www.example.com/

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)

Web Browser
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/SSL = 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

https://example.com/

Firewall (or attacker)

Log files or other servers

TLS-encrypted web page with CORRECT certificate

DNS Server

example.com?

10.1.1.123

Dnskey

Rrsigs

TlSA

Web Browser w/DANE

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.
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
  • ?
DNSSEC Deployment In Asia
ccTLD DNSSEC Status on 2013-03-25

Map courtesy of Shinkuro, Inc.
Panel Discussion
Our Panel

Moderator:
- **Dan York**, Senior Content Strategist, Internet Society

Panelists:
- **Sanjeev Gupta**, Principal Technical Architect, DCS1 Pte
- **Jitender Kumar**, Technical Account Manager, Afilias
- **Richard Lamb**, DNSSEC Program Manager, ICANN
Next Steps In Deploying DNSSEC
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 Steps For Network Operators and Enterprises

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

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/
Thank You!