Securing Core Internet Functions – DNSSEC

Mark Kosters
ARIN CTO
Core Internet Functions: Routing & DNS

- The Internet relies on two critical resources
  - DNS: Translates domain names to IP addresses and IP addresses to domain names
  - Routing: Tells us how to get to an IP address

- These critical resources are not secure
- DNSSEC helps secure these critical resources
Securing DNS With DNSSEC
How DNS Works

Question: www.arin.net A

Resolver

Caching forwarder (recursive)

www.arin.net A

root-server

Ask net server @ X.gtld-servers.net (+ glue)

www.arin.net A?

gtld-server

Ask arin server @ ns1.arin.net (+ glue)

www.arin.net A?

arin-server

Add to cache

192.168.5.10
What is DNSSEC?

• A DNS extension which authenticates responses
  – When you ask how to get to www.arin.net, DNSSEC verifies the answer is from ARIN and not someone pretending to be us
• Doesn’t ensure the answer is correct, just that it’s coming from the right place
Why is DNSSEC Important?

• Standard DNS is not secure
  – Trivial to spoof (provide false responses)
  – ... so an attacker can redirect people looking for www.arin.net to his own site
  – ... and then steal login information.
• DNSSEC is (surprise) secure
  – An attacker can try to redirect traffic, but DNSSEC will show it’s not a valid response
DNS Cache Poisoning

- Attacker gives the nameserver a “poisoned” (incorrect) response to www.arin.net
- If accepted, this nameserver will direct people to the fake site, typically for hours
- ... and any nameservers that trust the poisoned one will also become poisoned.
Case Study: Kashpureff Attack

- Eugene Kashpureff didn’t like Internic’s control of top level domains
- In 1996, he used DNS cache poisoning to redirect Internic traffic to his own site
- Kashpureff was eventually convicted of computer fraud
- This attack could have been prevented with DNSSEC
Case Study: Kaminsky Flaw

- 2008: Dan Kaminsky discovered a fundamental flaw in the DNS protocol
  - 65,536 Transaction IDs in DNS makes it easy to guess the right one & spoof
- Updates to DNS software makes this flaw more difficult to exploit, but not impossible
- These attacks can be prevented with DNSSEC
Case Study: Bradesco

• Bradesco is a bank in Brazil
• DNS cache poisoning attack resulted in 1% of the bank’s customers being redirected to a fake site
  – Getting login credentials for 1% of a large bank’s customers could be disastrous
• **Networks not using DNSSEC are vulnerable to a similar attack**
Other Uses

1. Protect DKIM & SPF
   - Without DNSSEC, an attacker can make use your email addresses for spam.
2. SSH Initial Host Key Exchange
   - Protect SSH Fingerprint (SSHFP) records.
3. PGP Key Distribution
   - Use _pka records to distribute PGP keys easily usable by GnuPG
4. DANE
   - Coming standard from the IETF to use DNS as a global public key infrastructure.
## DNSSEC Usage Statistics

<table>
<thead>
<tr>
<th></th>
<th>ARIN 39</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Orgs with DNSSEC</td>
<td>139</td>
</tr>
<tr>
<td>Total Number of Delegations</td>
<td>620,412</td>
</tr>
<tr>
<td>DNSSEC Secured Zones</td>
<td>671</td>
</tr>
<tr>
<td>Percentage Secured</td>
<td>0.11%</td>
</tr>
</tbody>
</table>
How Can We Increase Usage?

• Identify Barriers to Adoption
  – Don’t know how
  – Don’t have time
  – “… but I haven’t been attacked”

• Then, knock them down
  – Increased awareness
  – Education/assistance
  – “An ounce of prevention is worth a pound of cure”
Using DNSSEC with ARIN

• **Remember:** this is for reverse DNS, not forward DNS

• Use your DNS server software to:
  – Generate your key pair
  – Create DS records to upload to ARIN via ARIN Online or Reg-RWS
  – Sign your DNS zones
Completing DNSSEC Configuration

• Ensure the required DNSKEY, RRSIG, NSEC, and DS records are published in your nameservers
  – Consult your zone file.....

• ARIN provides only reverse DNSSEC
  – Make sure to also secure your forward DNS through your domain registrar
How It Works

- DNSSEC adds new resource records into your zone file.
- These records are signed off-line.
- Two types of public/private key pairs
  - Zone Signing Key (ZSK) is used to sign records in the zone
  - Key Signing Key (KSK) signs the ZSK. Usually longer lived than the ZSK.
What a Unsigned Zone Looks Like

<table>
<thead>
<tr>
<th>Zone Label</th>
<th>TTL</th>
<th>Type</th>
<th>Name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>SOA</td>
<td>ns1.arin.net</td>
<td>dns-ops.arin.net. 2016072520 10800 3600 604800 3600</td>
</tr>
<tr>
<td>0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>NS</td>
<td>ns1.arin.net</td>
<td></td>
</tr>
<tr>
<td>0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>NS</td>
<td>ns2.arin.net</td>
<td></td>
</tr>
<tr>
<td>0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>NS</td>
<td>ns2.lacnic.net</td>
<td></td>
</tr>
<tr>
<td>0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>NS</td>
<td>secl.apnic.net</td>
<td></td>
</tr>
<tr>
<td>0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>NS</td>
<td>secl.authdns.ripe.net</td>
<td></td>
</tr>
<tr>
<td>1.0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>PTR</td>
<td>host-199-43-0-1.arin.net</td>
<td></td>
</tr>
<tr>
<td>10.0.43.199.in-addr.arpa</td>
<td>10800</td>
<td>PTR</td>
<td>host-199-43-0-10.arin.net</td>
<td></td>
</tr>
</tbody>
</table>
A signed zone file will have RRSIG, NSEC, and DNSKEY records.
DNSSEC Adds New Record Types

**DNSKEY** – records holding the public zone signing key and key signing key

**RRSIG** – records holding the cryptographic signatures of the other DNS records

**NSEC** – records cryptographically stitching the other records together

**DS** – these point to your zone like an NS record *(needed in the parent zone)*
How Do I Know It’s Working?

Use a DNSSEC validating resolver.

Popular options include:

- [www.internetsociety.org/deploy360/dnssec/](http://www.internetsociety.org/deploy360/dnssec/)
How to Test

• DNS Viz gives a good presentation
  http://dnsviz.net/d/0.43.199.in-addr.arpa/dnssec/

• Give it a shot!
Initial Takeaways

• If you’re not using DNSSEC, you’re vulnerable to a DNS cache poisoning attack

• Plenty of readily available documentation regarding implementation details
A Really Easy Way to Use DNSSEC as a User

• Use google public DNS for your resolver
• It’s cloud computing!
• It has DNSSEC turned on by default
Thoughts About Cloud Computing in General

• From a user:
  – You may need to provide sensitive data
  – Cloud resources are ephemeral
  – How do you know you are going to the right place?
  – How can you tell afterwards?
Thoughts About Cloud Computing in General...

• From a content provider
  – Given this is a shared system:
  – How do you know you have the right connections for sensitive info (data/payments)?
  – How does the cloud provider ensure isolation between clients?
Q&A