The need for DNSSEC

DNS has weak inbuilt security

- ID field, 16 bit integer, returned in reply.
  - Some implementation use 14 bits
  - Multiple queries allow “birthday attack”
  - 16,384 packets is not a lot

- UDP generally preferred over TCP
  - No source address validation
  - Authoritative server addresses well known

- Spoofing data is hard to detect
  - Not much monitoring of DNS server caches
  - Increasingly targeted
DNSSEC adds security to DNS

- Authoritative server replies now signed.
  - Queries **not** signed - one way security.

- Keys published in zones like other data.
  - New DNS RR types for keys, signatures (and others) specific to DNSSEC.

- All sorts of usual stuff
  - Expiry dates for keys and signatures
  - Key rollover mechanisms
  - Support for different algorithms
Signatures

New DNS resource record RRSIG

- Sent automatically to DNSSEC aware resolvers
  - Flagged by setting D0 bit in query
- One per RRSET
  - RRSET has same owner, class and type
- Not used for NS records (more on that later)

<table>
<thead>
<tr>
<th>$ORIGIN internet.co.uk.</th>
</tr>
</thead>
<tbody>
<tr>
<td>@ SOA ...</td>
</tr>
<tr>
<td>RRSIG SOA ...</td>
</tr>
<tr>
<td>www A ...</td>
</tr>
<tr>
<td>A ...</td>
</tr>
<tr>
<td>A ...</td>
</tr>
<tr>
<td>RRSIG A ...</td>
</tr>
</tbody>
</table>
New DNS resource record DNSKEY

- Two types of keys (convention not protocol):
  - Zone Signing Keys (ZSKs) - used to sign zone data
    - short, fast signature verification, short lifetime
  - Key Signing Keys (KSKs) - used to sign KSKs
    - long, long signature verification, long lifetime

$ORIGIN internet.co.uk.

@ DNSKEY 256 3 5 ( AQOeiiR0GOMYkDshWoSKz9XzfwJr1AYtsmx3TGkJaNXVbfi/2pHm822aJ5iI9BMzNXxeYCMzZRD99WYwYqUSdjMmmAphXdvxegXd/M5+X7OrzKBaMbCvdlFLUUh6DhweJBjEVv5f2wwjM9XzcnOf+EPbtG9DMBmADjFDCzw/rjwvFw== ) ; key id = 60485
Delegations

Trust passes from parent and child zones

• Reminder on delegation data
  – Child is authoritative not parent
  – If NS records disagree then child wins
  – Parent data is just a hint

• DNSSEC handles delegations to fit these principles
  – NS records are not signed
  – New DNS resource record - DS (Delegation Signer)
    • Hash of child DNSKEY record data
    • Signed itself by an RRSIG

• Passes right way up to the root zone
  – Root zone keys must be implicitly trusted.
DNSSEC made easy
The chain of trust

$ORIGIN co.uk.
internet internet DS ... 2
internet RRSIG DS ...

$ORIGIN internet.co.uk.
@ DNSKEY ... 1
@ DNSKEY ... 2
@ RRSIG DNSKEY ... 2
Provable non-existence

Two new DNS resource records - NSEC and NSEC3

- Define a span - two adjacent existing names
  - Zone file contains `aaa` and `ccc`, client asks for `bbb`
  - Server responds with NSEC for `aaa` to `ccc`
  - Proves that `bbb` does not exist

```
$ORIGIN internet.co.uk.

aaa  A ...  
RRSIG A ...
NSEC  ccc ...
RRSIG  NSEC ...

ccc  A ...
```
Recent RFCs

Tackle implementation issues

- Zone file walking
  - Using NSECs can walk a zone file
  - If privacy is not an issue then bandwidth is!
- NSEC3 used instead of NSEC where needed
  - Spans of hashed names
- Huge increase in zone file size
  - Immediate 10x size increase
- Opt-out allows choice of signed delegations
  - No child key no security on delegation
  - Allows organic zone file growth
- Not quite finished - Automated root zone key rollover
DNSSEC made easy

The practice
DNSSEC made easy

Using secured incoming DNS data

Putting into practice simpler than understanding theory

• Caveat - Not all of this is possible yet

• Securing caching resolvers
  – Find and install root zone keys (if only!)
  – Turn on DNSSEC
  – Done !!

• Securing applications at the OS level
  – Turn on DNSSEC in resolver library
    • Backwards compatibility - Use DNSSEC if present, otherwise work as before. (Now)
    • Strict DNSSEC - Only use DNSSEC, unsigned records discarded. (5 years?)
Securing outgoing DNS data

This requires planning

• Generate keys
  – Choices on key sizes - KSKs, ZSKs, size etc
  – Choices on securing keys - HSMs, silo keys etc

• Sign the zones
  – Choices on mechanism - crypto accelerators
  – Choices on signature lifetimes - resigning timetable
  – Choices on delegations - sign all or opt-out

• Resource planning
  – 10x zone file increase
  – Higher bandwidth
  – More TCP to nameserver

• Send keys to registry

DNSSEC made easy
Best practice tips

We are writing documents on this!

- Signing schedules
  - Ensure always a current signature
  - Match zone generation/reload schedule
  - Implement **continuous signing** if zones not reloaded

- Ensure always active keys
  - Key rollover strategy
  - Schedule transmission of keys to registry

- If you delegate zones as well
  - Mechanism for receiving keys
  - Manage growth of zones
DNSSEC made easy
The politics
Some early adopters

- Isolated trust anchors
  - Individual registries have signed their zones
    - .se, .pr, RIPE
  - Sysadmins must manually find and install keys
  - No automated key rollover - manual process

- Does not scale
  - Whole point of DNS is a single root!

- Others insistent they will not sign yet
  - .uk, .de - zone walking solution
  - .com - opt-out
Two different camps on signing the root

• Camp one - the ‘hidden agenda’ brigade
  – US DoC will have too much control
  – Signatures have a special meaning
  – Needs a new body to manage root signing

• Camp two - the ‘just get on with it’ brigade
  – US DoC already has control - changes nothing
  – Signatures are just error checking
  – IANA and RZM (Verisign) already control this

• Where is this going?
  – Root politics already difficult
  – IANA now ready to do this (taking over RZM function?)
  – US DoC NTIA consulting on way forward
Remember

- DNSSEC is coming
  - Internet must be secured in layers - DNS layer is critical
- Protocol is a lot to learn but straightforward
- Implementation has two parts
  - Securing incoming DNS data - simple
  - Securing outgoing DNS data - requires planning
- And by the time you are ready
  - They might have signed the root!
DNSSEC made easy

The end

Questions?

jay@nominet.org.uk