Introduction to DNSSEC

Dr. Jeroen van der Ham
<jeroenh@randomdata.nl>
Warning

- Cryptography is hard
- Cryptography is even harder to do right
Agenda

- Introduction
- DNS Resolving
- Cryptography
- DNSSEC
About me

- ’96–’02 Cognitive Artificial Intelligence UU
- ’03–’04 System and Network Engineering UvA
- ’04–’10 PhD Research at SNE group UvA
- ’10–’13 Post-Doc at SNE group
- ’13- Lab Teacher at SNE Master
DNSSEC

- DNSSEC provides mechanism to establish integrity and authenticity of data
- DNSSEC does not provide encryption
DNS Resolving

- Caching Nameserver
- Authoritative Nameserver
- Local Resolver
- /etc/resolv.conf
- /etc/hosts
DNS Resolving

- DNS Root
- .nl Nameserver
- randomdata.nl Nameserver
- Caching Nameserver
DNS Resolving

Caching Nameserver

- DNS Root
- .nl Nameserver
- randomdata.nl Nameserver
Delegation

google.com.
com.
net.
org.
nl.

overheid.nl.
google.nl.
randomdata.nl.
Cryptography

Shared Key

Internet

Shared Key

Sender: Thomas

Receiver: Peter

credits: cisc Univ. of Hong Kong
Cryptography

Sender: Thomas

Dear Peter,
This is our Company's proposal.

Peter's Public key

Internet

Receiver: Peter

Dear Peter,
This is our Company's proposal.

Peter's Private key

credits: cisc Univ. of Hong Kong
Certificate Authority

User: Thomas
Thomas's Private key

User: Peter
Peter's Private key

Keeper: Miss C.A.

Thomas's Public key
Peter's Public keys

credits: cisc Univ. of Hong Kong
DNSSEC Ingredients

- **DNSKEY** – Public Key for a zone
- **DS** – Delegation Signer, hash of a DNSKEY
- **RRSIG** – Resource Record Signature
Secure Delegation

DNSKEY .
  .  DS nl.
    DNSKEY nl.
      DS randomdata.nl.
        DNSKEY randomdata.nl.
        randomdata.nl.
Tree-Walking

• Validating delegation from “.” (root) to “nl.” zone

1. Signing key of root zone is known to the resolver
2. Ask “.” zone for DS of “nl.” zone
3. Ask “nl.” zone for DNSKEY of “nl.” zone
4. Verify that DS contains a valid hash of DNSKEY
Missing Steps

• Validating delegation from “.” (root) to “nl.” zone
  1. Signing key of root zone is known to the resolver
  2. Ask “.” zone for DS of “nl.” zone
  3. Ask “nl.” zone for DNSKEY of “nl.” zone
  4. Verify that DS contains a valid hash of DNSKEY
  5. Get the signatures of the DS record
  6. Check that the DS record for “nl.” authentic
Manual Tree-Walking

1. dig @a.root-servers.net. . dnskey
2. dig @a.root-servers.net. nl. ds
3. dig @ns1.dns.nl. nl. dnskey
4. Verify hash ds – dnskey
5. dig +dnssec @a.root-servers.net nl. ds
Even more steps!

- 3a. Get DNSKEY for .nl -> Zone signing key
- 3b. Get signed DNSKEY for .nl -> Key signing key
RRSIG Example

- host.example.com. 86400 IN RRSIG A 5 3 86400 20030322173103 ( 20030220173103 2642 example.com. oJB1W6WNGv+ldvQ3WDG0MQkg5IEhjRip8WTr PYGv07h108dUKGMeDPKijVCHX3DDKdfb+v6o B9wfuh3DTJXUAfI/ M0zm0/zz8bW0Rznl803t GNazPwQKkRN20XPXV6nwwfoXmJQbsLNrLfkG J5D6fwFm8nN+6pBzeDQfsS3Ap3o= )
<table>
<thead>
<tr>
<th>Part</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type covered</td>
<td>Record type this RRSIG is about</td>
</tr>
<tr>
<td>Algorithm</td>
<td>Signature algorithm used; 5 is RSA/SHA-1</td>
</tr>
<tr>
<td>Labels</td>
<td>Number of labels of owner (without root)</td>
</tr>
<tr>
<td>TTL</td>
<td>Original time to live</td>
</tr>
<tr>
<td>Expiration and Inception</td>
<td>Signature validity date bounds</td>
</tr>
<tr>
<td>Key Tag</td>
<td>To help find(^2) (not identify) the signing key</td>
</tr>
<tr>
<td>Signer’s Name</td>
<td>Owner name of zone and key to use</td>
</tr>
<tr>
<td><strong>Signature</strong></td>
<td>Signature (in Base64)</td>
</tr>
</tbody>
</table>

\(^2\)In fact it is a kind of checksum on the RDATA of the DNSKEY RR
DNSKEY Example

- example.com. 86400 IN DNSKEY 256 3 5
  ( AQPSKmynfzW4kyBv015MUG2DeIQ3 Cbl +BBZH4b/0PY1kxkmvHjcZc8no kfzj31GajIQKY+5CptLr3buXA10h WqTkF7H6RfoRqXQeogmMHfpftf6z Mv1LyBUgia7za6ZEzOJB0ztyvhjL 742iU/ TpPSEDhm2SNKLIjfUppn1U aNvv4w== )
<table>
<thead>
<tr>
<th>Part</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flags</td>
<td>Zone key (KSK+ZSK); Secure entry point (KSK)</td>
</tr>
<tr>
<td>Protocol</td>
<td>Always 3 (for backward compatibility with KEY RR)</td>
</tr>
<tr>
<td>Algorithm</td>
<td>Signature algorithm used; 5 is RSA/SHA-1</td>
</tr>
<tr>
<td>Public key</td>
<td>Key used for signing (in Base64)</td>
</tr>
</tbody>
</table>
DS Example

dskey.example.com. 86400 IN DNSKEY 256 3 5 ( AQOeiiR0GOMYkDshWoSKz9Xz
    fwJr1AYtsmx3TGkJaNXVbfi/
    2pHmS22aJ5i19BmzNxxeYcmZ
    DRD99YWvUSdJmmMphXdvx
    egXd/M5+X70rzKBaMbCVdFLU
    Uh6DhweJBjEVv5f2wwjM9Xzc
    n0f+EpbtG9DMBmADjFDc2w/r
    ljwvFw==
    ) ; key id = 60485

dskey.example.com. 86400 IN DS 60485 5 1 ( 2BB183AF5F22588179A53B0A98631FAD1A292118 ) ; SHA-1

dskey.example.com. 86400 IN DS 60485 5 2 ( D4B7D520E7BB5F0F67674A0C
    CEB1E3E0614B93C4F9E9983
    83F6A1E4469DA50A ) ; SHA-256
<table>
<thead>
<tr>
<th>Part</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Tag</td>
<td>To help find (not identify) the signing key</td>
</tr>
<tr>
<td>Algorithm</td>
<td>Signature algorithm of the signing key</td>
</tr>
<tr>
<td>Digest Type</td>
<td>Hashing algorithm used; 1 is SHA-1, 2 is SHA-256</td>
</tr>
<tr>
<td>Digest</td>
<td>Sequence of case-insensitive hexadecimal digits</td>
</tr>
</tbody>
</table>
Absence Proof

- NSEC – authenticated denial of existence
  - uses canonical ordering
    - bert.tld 100 NSEC ernie.tld. A RRSIG TXT NSEC
- NSEC3
  - uses hashes of domain names
## NSEC3 record content

<table>
<thead>
<tr>
<th>Part</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hash Algorithm</td>
<td>1 for SHA-1</td>
</tr>
<tr>
<td>Flags</td>
<td>Opt-out for skipping unsigned delegations</td>
</tr>
<tr>
<td>Iterations</td>
<td>Number of times hash is calculated</td>
</tr>
<tr>
<td>Salt Length</td>
<td>0-255 (not represented in text format)</td>
</tr>
<tr>
<td>Salt(^3)</td>
<td>Prevents dictionary attacks</td>
</tr>
<tr>
<td>Hash Length</td>
<td>1-255 (not represented in text format)</td>
</tr>
<tr>
<td>Next Hashed Owner Name</td>
<td>(binary data)</td>
</tr>
<tr>
<td>Type Bit Maps</td>
<td>Types present at the current owner</td>
</tr>
</tbody>
</table>

\(^3\)Hash Algorithm, Flags, Iterations and Salt are communicated to slave servers via the NSEC3PARAM Resource Record in the zone apex.