

DNS Security

Secure Addressing and Key Distribution

Donald E. Eastlake 3rd

dee@cybercash.com

CyberCash, Inc.

<<http://www.cybercash.com>>

2100 Reston Parkway, Suite 430, Reston, VA 22091 USA

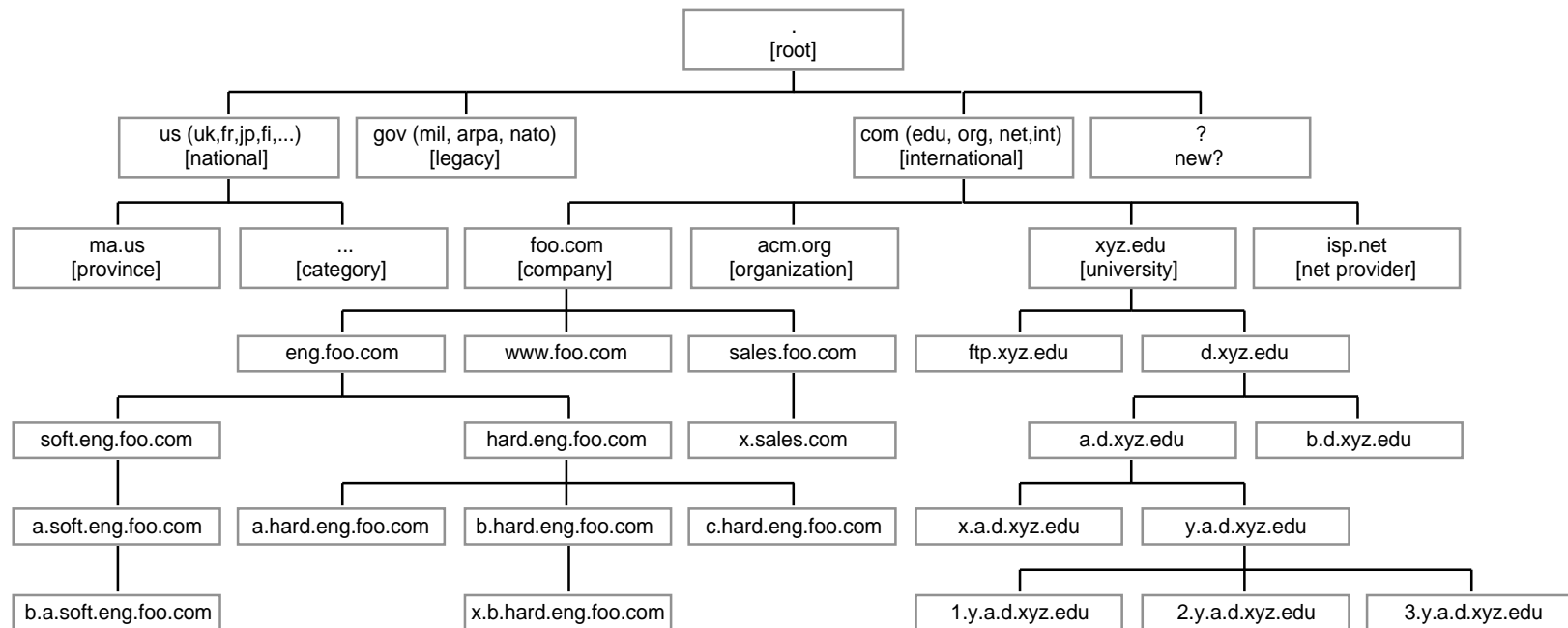
Topics

- What is the Domain Name System (DNS)?
 - DNS structure
 - domain names
 - resource records
- Domain Name System Security
- Secure Dynamic Update
- Questions?

DNS Structure

- A Hierarchy of named nodes
 - ., edu, mit.edu, ai.mit.edu, gnu.ai.mit.edu, ...
 - TLDs (top level domains) for all ISO 2 letter country codes and for COM, EDU, GOV, INT, MIL, & NET
- Nodes partitioned into zones
 - similar to UNIX directory structure and partition mounting
- Each zone has multiple servers whose names may be unrelated to the zone name
 - pothole.com's servers are: ns1.terra.net, ns2.terra.net, and ns.opal.com

DNS Zone Structure



Domain Name System Names

- Common Entity Names Are Already in the DNS
- Hosts: domain.name.tld
 - for routing, DNS, NTP, SMTP, IPSEC, etc.
- Users: account@domain.name.tld
 - for mail, ftp, IPSEC, etc., maps into account.domain.name
- IPv4 address: mapped into *.in-addr.arpa
- IPv6 address: mapped into *.ipv6.int
- Telephone number: mapped into *.tpc.int

DNS Resource Records

- Resource Records at each node
 - A, AAAA (X.25, ...) – addresses
 - MX – mail service forwarding
 - SOA, NS – zone structure
 - CNAME – aliasing
 - RP – responsible person
 - TXT – human readable text
 - LOC – geographic location
 - etc., etc.
 - (KEY, SIG, NXT – security)

Topics

- What is the Domain Name System (DNS)?
- Domain Name System Security
 - Features/Advantages
 - RR types
 - Documentation
 - Beta Implementation
- Secure Dynamic Update
- Questions?

Features of DNS Security

- Data origin authentication.
- Data does-not-exist authentication.
- Public key distribution.
- Transaction security.
- Update/other request authentication.

Advantages of DNS Security

- The DNS is deployed globally.
- Most entities you are interested in (hosts, users, IP addresses) are already mapped into the DNS.
- In many cases you get the key for free when you retrieve other information.
- Your resolver does the authentication for you.
- Short lived “certificates” (authenticated keys) are practical.

DNS Security

- Digitally signs DNS data providing data origin authentication and integrity.
- Provides for general public key storage and distribution.
- Optional transaction security.
- Optional query authentication for update or other uses.

DNS Security RR Types

- KEY- associates a public key, flags, etc., with a domain name
- SIG - authenticates an RRset, has time signed, expiration time, etc.
- NXT - specifies non-existent parts of name space and types which exist for a name.

Services Not Provided

- Data in DNS is public:
- No attempt to provide access control lists or other means to differentiate inquirers.
- No effort has been made to provide for any confidentiality for queries or responses.

DNS Security Documentation

- For main DNS Security document, see <<ftp://ftp.isdi.edu/draft-ietf-dnssec-secext-10.txt>>, approved as Proposed Standard but RFC not yet out.
- For DNS Security in update, see <<ftp://ftp.isi.edu/draft-ietf-dnssec-update-03.txt>>.
- Mailing list: dns-security@tis.com
 - to subscribe: dns-security-request@tis.com

DNS KEY RR Format

```

OWNER NAME, Type: KEY, Class (IN), TTL, RDSIZE,
               1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               flags                               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               protocol                           |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               algorithm                           |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               /
/                               public key                           /
/                               /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

For RSA Algorithm, public key=

```

+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| pub exp length|          public key exponent                      /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                                        /
+-              modulus                                          /
|                                                        /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

DNS KEY Flags

0	Don't use for authentication
1	Don't use for confidentiality
2	Experimental
3-4	(reserved)
5	User
6	Host/End-Entity
7	Zone
8	Reserved for IPSEC
9	Reserved for email
10-11	(reserved)
12-15	Signatory Field

DNS SIG RR Format

OWNER NAME, Type: SIG, Class (IN), TTL, RDSIZE,

										1	1	1	1	1	1	1	1	1	1	2		2		2		2		2		2		2		3		3	
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1						
type covered										algorithm										labels																	
original TTL																																					
signature expiration																																					
time signed																																					
key footprint															signer's name																						
signature																																					

Special Considerations for TTL

- If part of signature, count down of TTL by caching servers breaks signatures.
- If not part of signature, unscrupulous servers can set to arbitrary large value.
- Answer: put original TTL in SIG. This bounds value but keeps signatures valid.

Non-zone SIGs

- Transaction SIG signed by server host, not zone. Binds response to request.
 - resolver -> request -> server host
 - resolver <- response | SIGsrv <- server host
- Update SIGs authenticate request, signed with update key, not zone.
 - resolver -> request | SIGauth(s) -> server

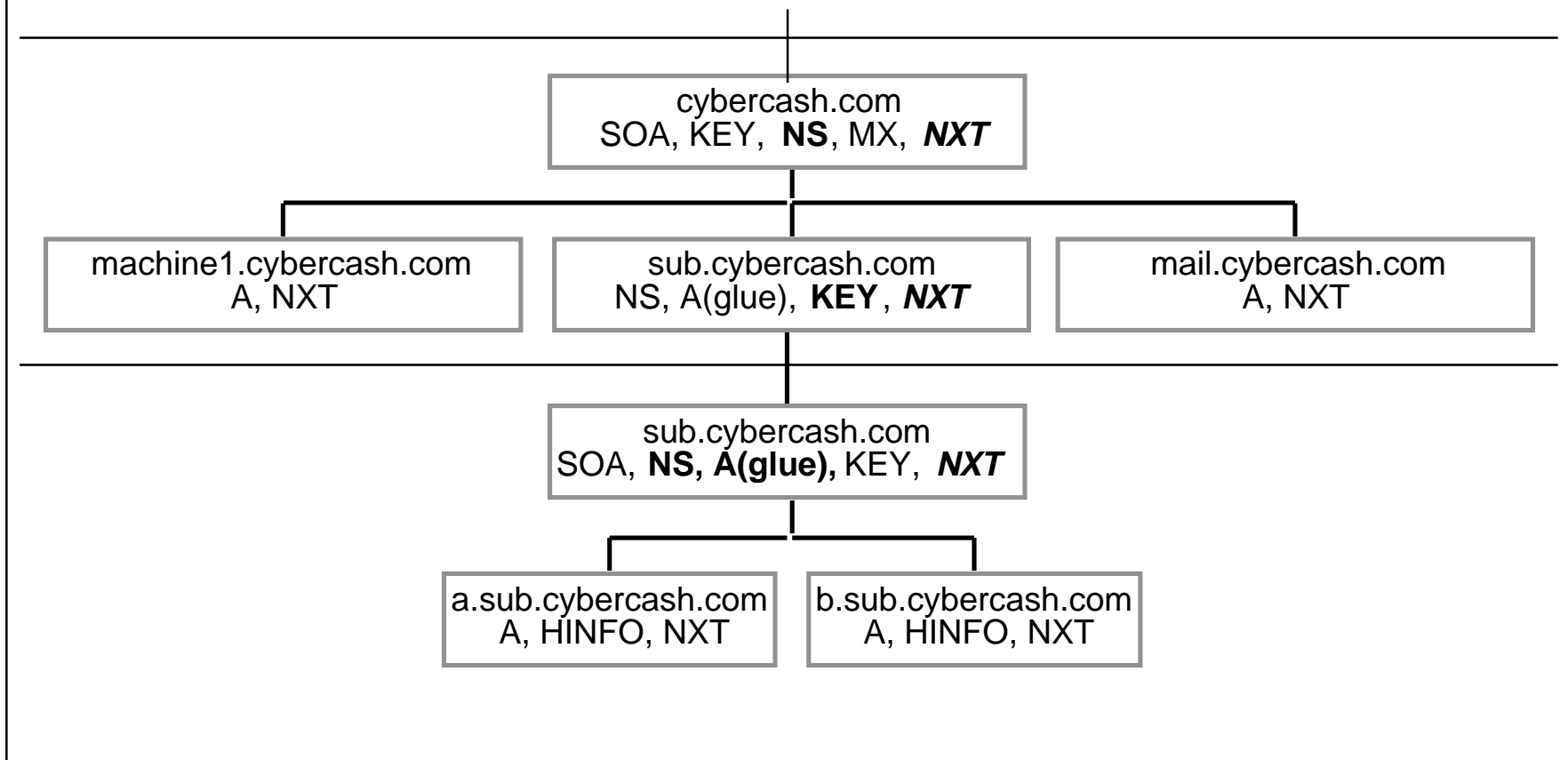
DNS NXT RR Format

```

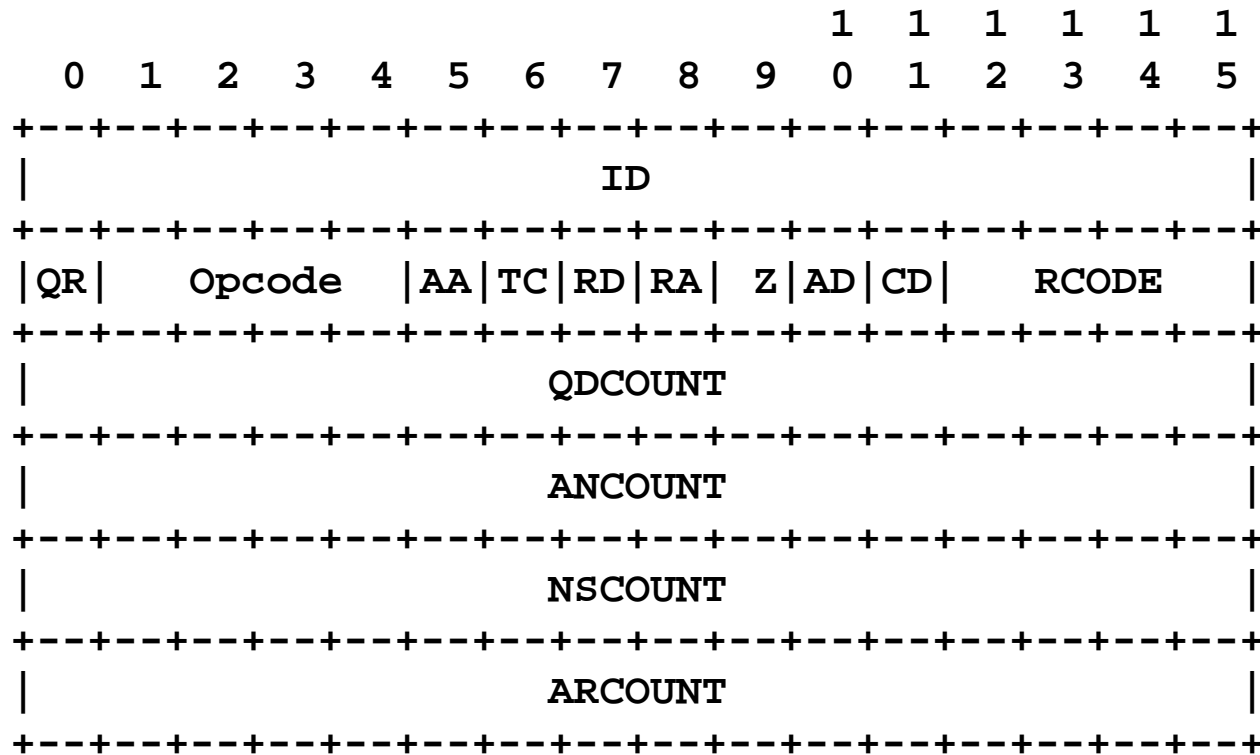
OWNER NAME, Type: NXT, Class (IN), TTL, RDSIZE,
                1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          next domain name                                          /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          type bit map                                             /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

Delegation Points



AD/CD Header Bits



AD = Authentic Data, CD = Checking Disabled

Problems with CNAME

- Non-security DNS server doesn't automatically provide SIGs but does automatically follow CNAME changes.
- Separate retrievals to a non-security server can not get the SIGs for a CNAME.
- Secure CNAMEs must be served by security aware servers.

Beta DNSSEC Code Available

- TIS/DNSSEC based on bind-4.9.3-REL
 - <ftp://ftp.tis.com/pub/DNSSEC/README>
 - <mailto:tisdnssec-support@tis.com>
- uses RSAREF (EuroRef outside of USA)
- comes with a security enabled DIG
 - <ftp://ftp.tis.com/pub/DNSSEC/sdig.sunos4.gz>
- first secure zone: sd-bogus.tis.com

Topics

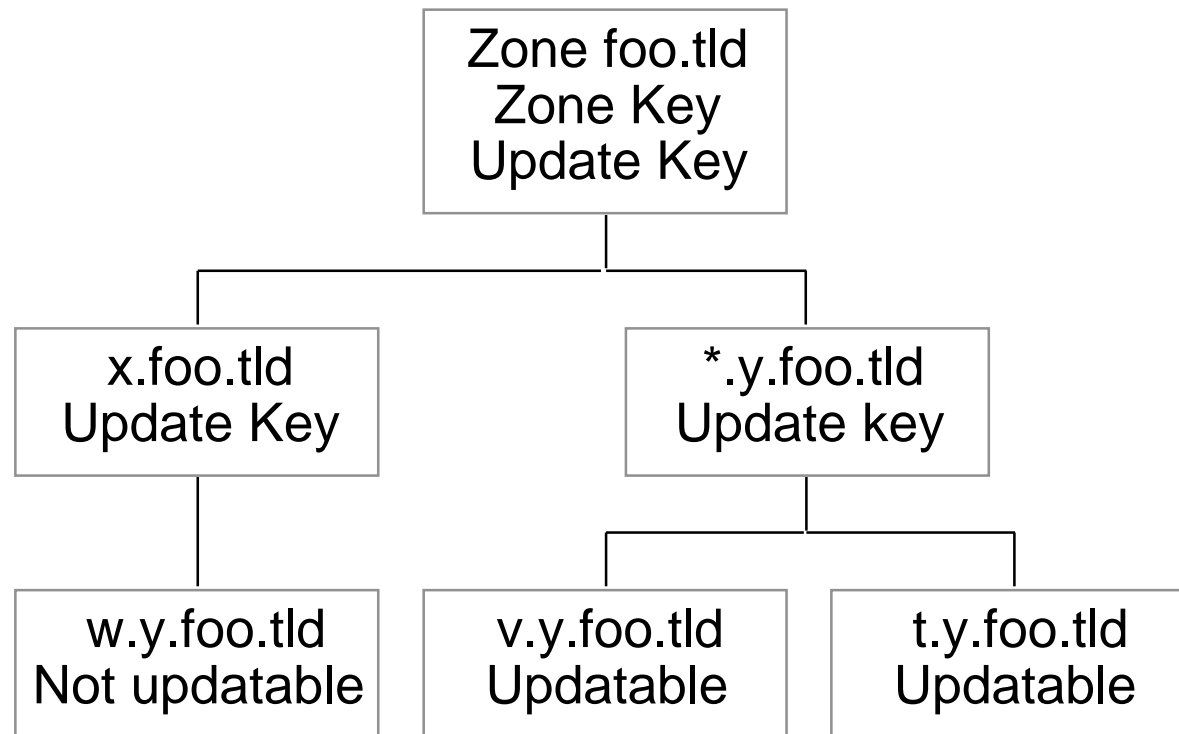
- What is the Domain Name System (DNS)?
- Domain Name System Security
- Secure Dynamic Update
- Questions?

Dynamic Secure Zone Modes

SUMMARY OF DYNAMIC SECURE ZONE MODES

CRITERIA:	MODE A	MODE B
-----+	-----+	-----
Zone Key	Off line	On line
-----+	-----+	-----
Server Workload	Low	High
-----+	-----+	-----
Static Data Security	Very High	Medium-High
-----+	-----+	-----
Dynamic Data Security	Medium	Medium-High
-----+	-----+	-----
Key Restrictions	Fine grain	Coarse grain
-----+	-----+	-----
Dynamic Data Temporality	Transient	Permanent
-----+	-----+	-----
Dynamic Key Rollover	No	Yes
-----+	-----+	-----

Update Key Hierarchy in a Zone



DNS KEY Signatory Flags

UPDATE KEY RR SIGNATORY FIELD BITS

12	13	14	15
+	+	+	+
zone	strong	unique	general
+	+	+	+

ZONE KEY RR SIGNATORY FIELD BITS

12	13	14	15
+	+	+	+
mode	strong	unique	general
+	+	+	+

mode=0 -> mode A, mode=1 -> mode B

The in-key.int. Domain

`<key-hash>.<key-footprint>.algorithm.in-key.int.`

`$ORIGIN xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xx`

`xx.xxxx.xxxx.xxxx.xxxx.1.in-key.int.`

`IN KEY <flags> 0 1 (45IkskceFGgiWCn/GxHhai6VAuHAoNUz4YoUMxF
cby9k/yvedMfQgKzhH5er0Mu/vILz80jEeC8aTrO+KKmCaY1tVfSCqQYn6/
/11U6Nld= ;key)`

`IN SIG KEY 1 3 (;type-cov=PTR, alg=1, labels=3
19991202030405 ;signature expiration
19951211100908 ;time signed
2143658709 ;key footprint
example.tld. ;signer`

`MxFcby9k/yvedMfQgKzhH5er0Mu/vILz45IkskceFGgiWCn/GxHhai6VAuH
AoNUz4YoU1tVfSCSsqQYn6//11U6Nld80jEeC8aTrO+KKmCaY=
;signature
)`

Miscellaneous

- DNS Boot File Additions
 - **pubkey** name flags protocol algorithm key-data
 - **keyfile** filename
- DNSSEC needs “secure time”
- It's is only one facet of Internet security

Topics

- What is the Domain Name System (DNS)?
- Domain Name System Security
- Questions?