DNSSEC Tutorial

Peter Losher

APNIC 32 – Busan, South Korea

August 28th, 2011

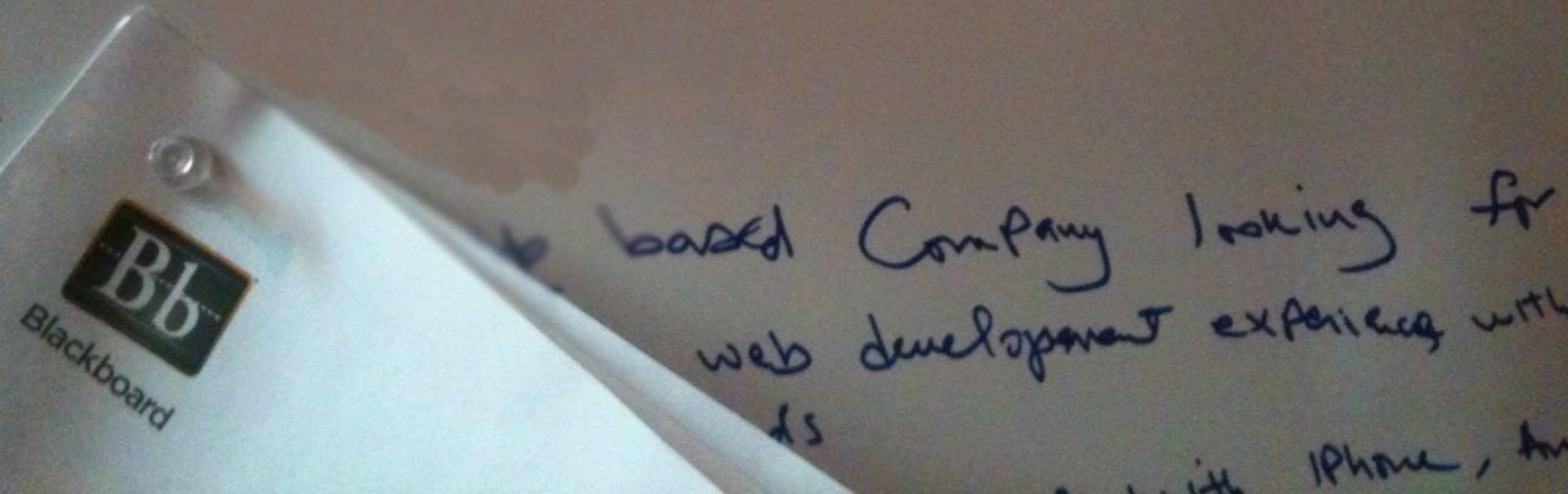
is the New "COBOL"

Deploy DNSSEC
NOW .... or something
bad might happen...

based Company looking for
web development experience with
ds
iPhone, and

is the New "COBOL"

Define a security standard for DNS that CAN be deployed, and operators will. ⟹ Deploy DNSSEC NOW .... or something bad might happen...



based Company looking for
web development experience with
ds
iPhone, An

About the Instructor

Peter Losher

Senior Operations Architect @ ISC

e-mail - plosher@isc.org

About ISC

- Internet Systems Consortium, Inc.
  - Headquartered in Redwood City, CA, USA
  - 501(c)(3) Nonprofit Corporation

- ISC is a public benefit corporation dedicated to supporting the infrastructure of the universal connected self-organizing Internet — and the autonomy of its participants — by developing and maintaining core production quality software, protocols, and operations.

DNS

What is DNS?

- Better question:

  How does Internet naming work?

- Computers like numbers, humans like names. DNS fixes that.

Addresses and Names

- An address is how you get to an endpoint
  - Typically there is a hierarchy:
    950 Charter Street, Redwood City, CA, 94063
    192.153.154.12, +1-919-271-8851
- A Name is how the endpoint is referenced
  - Usually no hierarchy:
    "Alan", "San Francisco", "ISC's website"

HOSTS.TXT

- Contained IP addresses and names of all the hosts on the Internet
- Contained some information about each host
- Distributed twice a week and available via FTP
  - Updates sent in via e-mail

HOSTS.TXT

```
; DoD Internet Host Table
;  30-Mar-93
;  Version number 1230
;
;
; The format of this file is documented in RFC 952, "DoD Internet
; as the file  rfc/rfc952.txt
;
; It may be retrieved via FTP using username "anonymous" with
; any password.
;
; The format for entries is:
;
; NET : NET-ADDR : NETNAME :
; GATEWAY : ADDR, ADDR : NAME : CPUTYPE : OPSYS : PROTOCOLS :
; HOST : ADDR, ALTERNATE-ADDR (if any): HOSTNAME,NICKNAME : CPUTYPE :
;   OPSYS : PROTOCOLS :
;
; Where:
;;  CPUTYPE = machine type (PDP-11/70, VAX-11/780, FOONLY-F3, C/30, etc.)
;;  OPSYS = operating system (UNIX, TOPS20, TENEX, ITS, etc.)
;;  PROTOCOLS = transport/service (TCP/TELNET,TCP/FTP, etc.)
;;  : (colon) = field delimiter
```

HOSTS.TXT

```
HOST : 128.105.5.1,128.105.2.250 : STAT.STAT.WISC.EDU,STAT.WISC.EDU : VAX-11/750 : UNIX :
    TCP/TELNET,TCP/FTP,TCP/SMTP,TCP/ECHO,TCP/FINGER,ICMP :
HOST : 128.109.140.1 : DUKE.CS.DUKE.EDU,CS.DUKE.EDU : SUN-3/180 : OS :
    TCP/TELNET,TCP/FTP,TCP/SMTP,UDP/DOMAIN :
HOST : 128.109.143.1 : UNCCVAX.UNCC.EDU : VAX-8530 : UNIX : TCP/TELNET,TCP/FTP,TCP/SMTP :
HOST : 128.109.153.1 : NCSUVX.NCSU.EDU : MICROVAX-II : UNIX :
    TCP/TELNET,TCP/SMTP,TCP/FINGER,UDP/DOMAIN :
HOST : 128.109.178.1 : NCSC.ORG : MICROVAX-II : UNIX : TCP/TELNET,TCP/FTP,TCP/SMTP :
HOST : 128.109.201.1 : ECSGATE.UNCECS.EDU : DEC-5000 : UNIX : TCP/TELNET,TCP/FTP,TCP/SMTP,UDP/DOMAIN :
HOST : 128.109.221.1 : VXA.UNCWIL.EDU : VAX : VMS ::
HOST : 128.110.190.1 : ANTHRO.ANTHRO.UTAH.EDU,ANTHRO.UTAH.EDU : SUN-4/60 : UNIX :
    TCP/FTP,TCP/SMTP,TCP/TELNET :
HOST : 128.110.198.1 : CTRSCI.MATH.UTAH.EDU : VAX-8600 : VMS : TCP/FTP,TCP/SMTP,TCP/TELNET :
HOST : 128.110.200.1 : COSMIC.PHYSICS.UTAH.EDU : DS-3100 : UNIX : TCP/FTP,TCP/SMTP,TCP/TELNET :
HOST : 128.112.128.1 : PRINCETON.EDU : SUN-4/330 : UNIX : TCP/TELNET,TCP/FTP,TCP/SMTP,UDP/DOMAIN :
```

HOSTS.TXT

- Drawbacks
  - Maintained by a single entity
  - Pulled (manually) from a single host
  - Namespace collisions
  - Consistency
- In 1983, it was determined that something had to be done

DNS RFCs

- 1983 - RFC 882 & 883
  - DOMAIN NAMES - CONCEPTS and FACILITIES
  - DOMAIN NAMES - IMPLEMENTATION and SPECIFICATION

DNS RFCs

Standards

[RFC974] C. Partridge. *Mail Routing and the Domain System*. January 1986.

[RFC1034] P.V. Mockapetris. *Domain Names — Concepts and Facilities*. November 1987.

[RFC1035] P. V. Mockapetris. *Domain Names — Implementation and Specification*. November 1987.

Proposed Standards

[RFC2181] R., R. Bush Elz. *Clarifications to the DNS Specification*. July 1997.

[RFC2308] M. Andrews. *Negative Caching of DNS Queries*. March 1998.

[RFC1995] M. Ohta. *Incremental Zone Transfer in DNS*. August 1996.

[RFC1996] P. Vixie. *A Mechanism for Prompt Notification of Zone Changes*. August 1996.

[RFC2136] P. Vixie, S. Thomson, Y. Rekhter, and J. Bound. *Dynamic Updates in the Domain Name System*. April 1997.

[RFC2671] P. Vixie. *Extension Mechanisms for DNS (EDNS0)*. August 1997.

[RFC2672] M. Crawford. *Non-Terminal DNS Name Redirection*. August 1999.

[RFC2845] P. Vixie, O. Gudmundsson, D. Eastlake, 3rd, and B. Wellington. *Secret Key Tr... DNS (TSIG)*. May 2000.

[RFC2930] D. Eastlake, 3rd. *Secret Key Establishment for DNS (TK...*

[RFC2931] D. Eastlake, 3rd. *DNS Request and Transa...*

[RFC3007] B. Wellington. *Secure Domain ...*

[RFC3645] S. Kwan, P. Garg, ... *Secret K...*

DNS S...

...yed DNS Software.. October

...ber. *Common DNS Implementation Errors and*

...ber Arithmetic*.* August 1996.

... Jinmei. *Common Misbehaviour Against DNS Queries for IPv6 Addresses*. May 2005.

...d Types

...1183] C.F. Everhart, L. A. Mamakos, R. Ullmann, and P. Mockapetris. *New DNS RR Definitions*. October 1990.

[RFC1706] B. Manning and R. Colella. *DNS NSAP Resource Records*. October 1994.

[RFC2168] R. Daniel and M. Mealling. *Resolution of Uniform Resource Identifiers using the Domain Name System*. June 1997.

[RFC1876] C. Davis, P. Vixie, T., and I. Dickinson. *A Means for Expressing Location Information in the Domain Name System*. January 1996.

[RFC2052] A. Gulbrandsen and P. Vixie. *A DNS RR for Specifying the Location of Services.*. October 1996.

[RFC2163] A. Allocchio. *Using the Internet DNS to Distribute MIXER Conformant Global Address Mapping ...* 1998.

[RFC2230] R. Atkinson. *Key Exchange Delegation Record for the DNS*. October 1997

[RFC2536] D. Eastlake, 3rd. *DSA KEYs and SIGs in the Domain Name S...*

[RFC2537] D. Eastlake, 3rd. *RSA/MD5 KEYs and SIGs in the ...*

[RFC2538] D. Eastlake, 3rd and O. Gudmundss... 1999.

[RFC2539] D. Eastlake, ...

[RFC2540] D. ...

...

... October 1989.

...tion*.* March 1994.

...less IN-ADDR.ARPA Delegation*.* March 1998.

...AB Technical Comment on the Unique DNS Root*.* May 2000.

..., L. Brunner-Williams, and B. Manning. *Domain Name System (DNS) IANA ...ations*. September 2000.

...perations

[RFC1033] M. Lottor. *Domain administrators operations guide.*. November 1987.

[RFC1537] P. Beertema. *Common DNS Data File Configuration Errors*. October 1993.

[RFC1912] D. Barr. *Common DNS Operational and Configuration Errors*. February 1996.

[RFC2010] B. Manning and P. Vixie. *Operational Criteria for Root Name Servers*.. October 1996.

[RFC2219] M. Hamilton and R. Wright. *Use of DNS Aliases for Network Services*.. October 1997.

Internationalized Domain Names

[RFC2825] IAB and R. Daigle. *A Tangled Web: Issues of I18N, Domain Names, and the Other Internet protocols*. May 2000.

[RFC3490] P. Faltstrom, P. Hoffman, and A. Costello. *Internationalizing Domain Names in Applications (IDNA)*. March 2003.

[RFC3491] P. Hoffman and M. Blanchet. *Nameprep: A Stringprep Profile for Internationalized Domain Names*. March 2003.

[RFC3492] A. Costello. *Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)*. March 2003.

BIND 9 Administrator Reference Manual

Bibliography and Suggested Reading

http://www.isc.org/sw/bind/arm96/Bv9ARM.ch09.html#rfcs

DNS Reality

- Globally distributed, loosely coherent, scalable, dynamic database

- Comprised of three components
  - "Name Space"
  - Servers making that space available
  - Resolvers (clients) which query the servers about the namespace

So, What is DNS?

- When asked for the address of `www.isc.org`, the distributed database known as DNS provides the answer:

`www.isc.org` ➔ `204.152.184.88`

- The database also contains the inverse "mapping"

What is DNS?

- DNS can contain additional information as well:
  - Addresses (ipv4, ipv6)
  - Security key information
  - Mail exchange information
  - Text records
    - An overly abused record type
  - Entertaining host information

What is DNS?

- DNS also provides negative results:

`mmm.isc.org` ➔ `NXDOMAIN`

- No such object in the database

What is DNS?

- NOTE:
  - No such object in the database
    is different from
  - No data of requested type for that object

`www.isc.org (hinfo)` → `null,NOERROR`

Global Distribution

- Data is maintained locally but retrievable globally
  - No single point of failure
- DNS lookups (queries) can come from any device
- Remote DNS data is locally cacheable to improve performance

Loose Coherency

- Database is always internally consistent
  - Every component (zone) has a serial number
  - Serial numbers are incremented on data change
- Changes to the master database are replicated to slave servers
- Cached data expires according to timeouts set by administrators

Scalable

- No limit to the size of the database
  - Single servers known to hold over 60,000,000 labels (`.com`)
- No limit to the number of queries
  - 24,000 queries per second not uncommon on "off-the-shelf" hardware
- Queries are automatically distributed between master, slave and cache servers

Reliable

- Data is replicated
  - Masters transfer data to slaves
- Clients can query master, slave or caching servers
- DNS uses UDP or TCP
  - DNS protocol deals with UDP retransmission, sequencing, etc.

Dynamic

- Database can be updated dynamically
  - Add/delete/modify any record
- Modification of master triggers replication
  - Only master can be dynamically updated

Namespace

- Hierarchical namespace
- Choose names based on
  - Location
  - Owner
  - Object

Namespace

- Fully qualified domain name (FQDN)

```
www.isc.org.
```

- DNS provides mapping from the FQDN to other resources
- Names are used as database keys

Domains and Domain Names

- A domain name consists of one or more labels or words
  - Labels are separated with '.'
  - Labels have at most 63 characters
- An FQDN has at most 255 characters, including dots

Domains and Domain Names

- Host names consist of the characters a-z, 0-9, and ‘-’
  - Other characters are allowed in labels
  - Specifically, "_" is a special character
- Name resolution is case insensitive

Resource Records

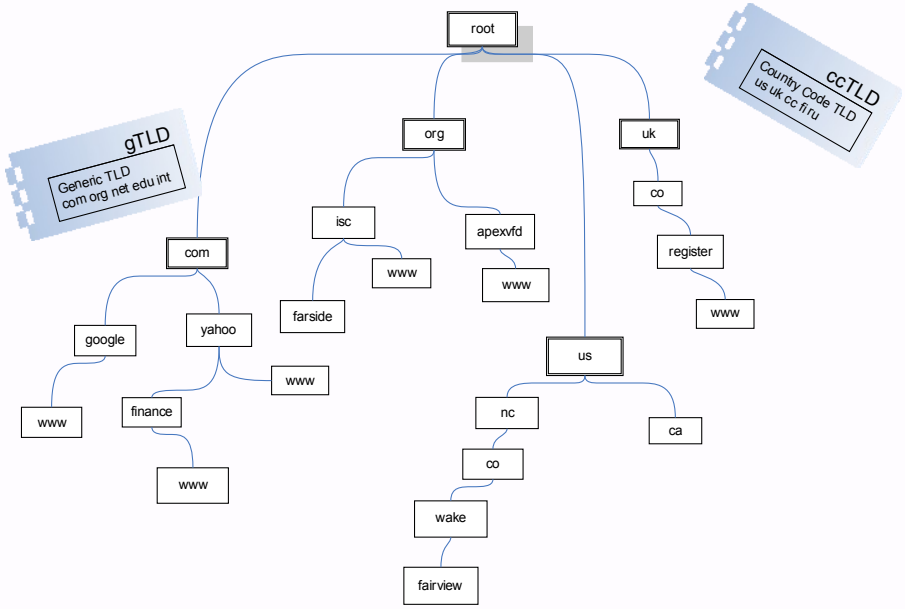
- The DNS maps those names into data called Resource Records (RRs)

`www.isc.org.` ➔ `A 204.152.184.88`

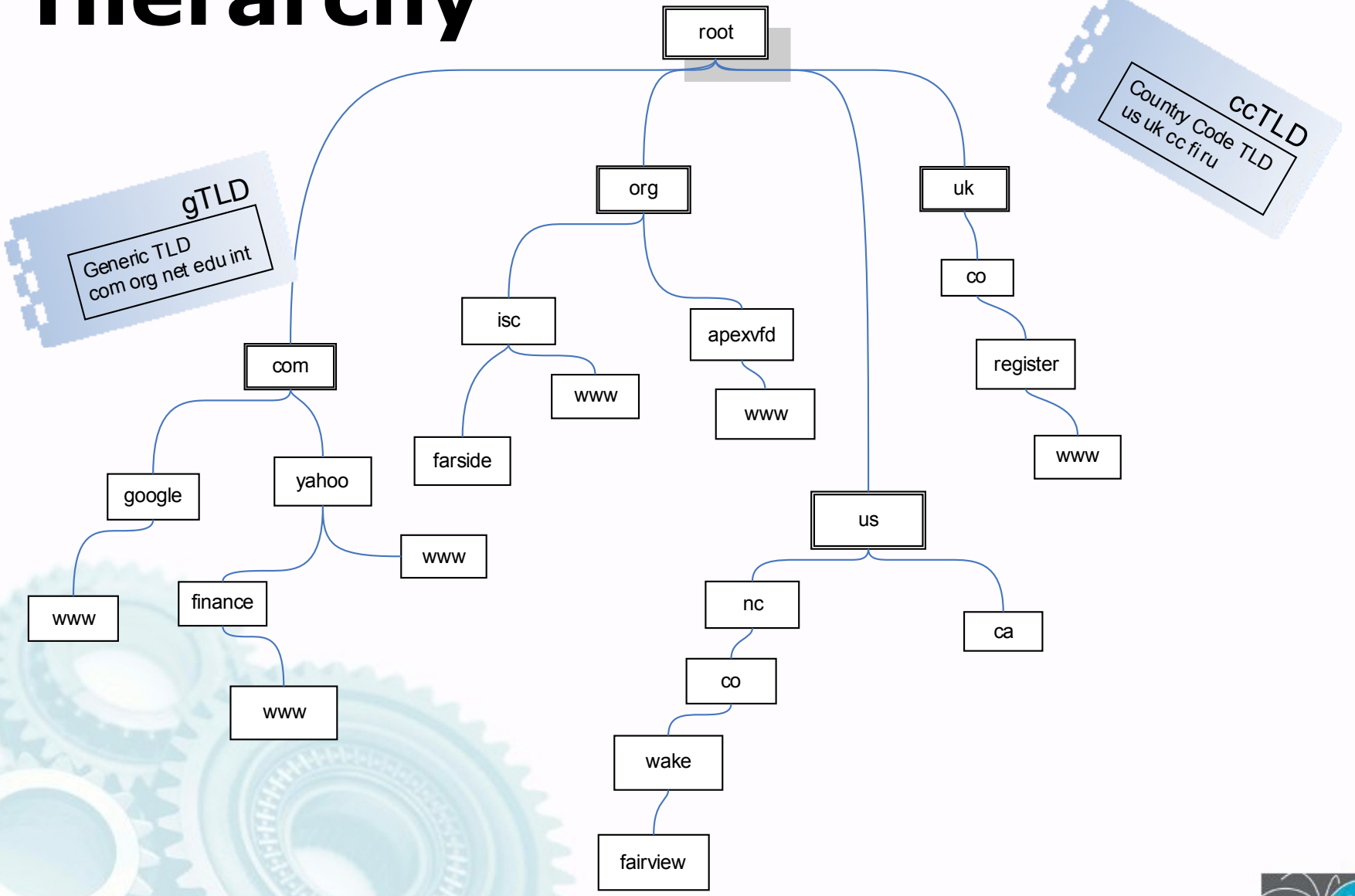
- This name has this address

Hierarchy

- Domain names map to a hierarchy chart
- root "." at top
- Chart splits at every dot
- No restriction to the number of branches

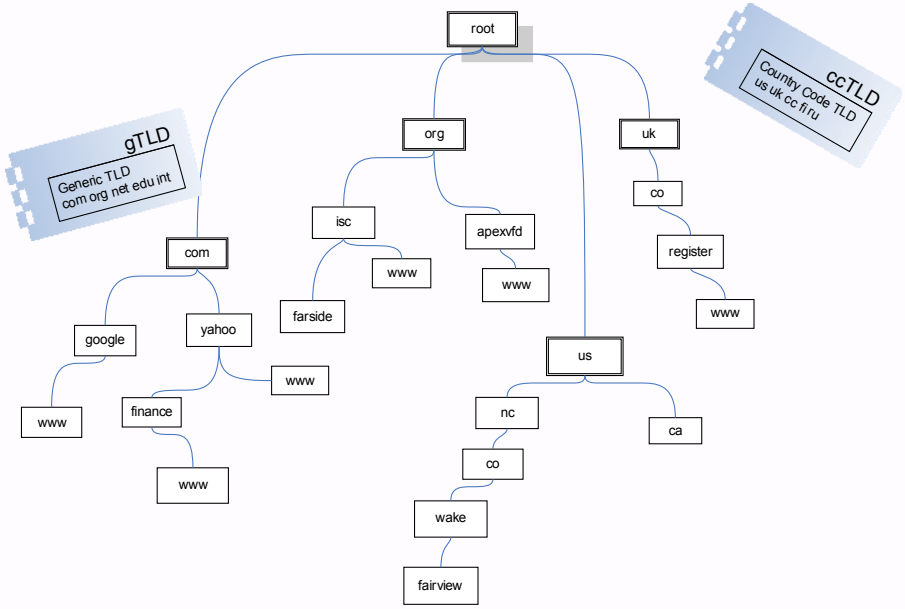


Hierarchy

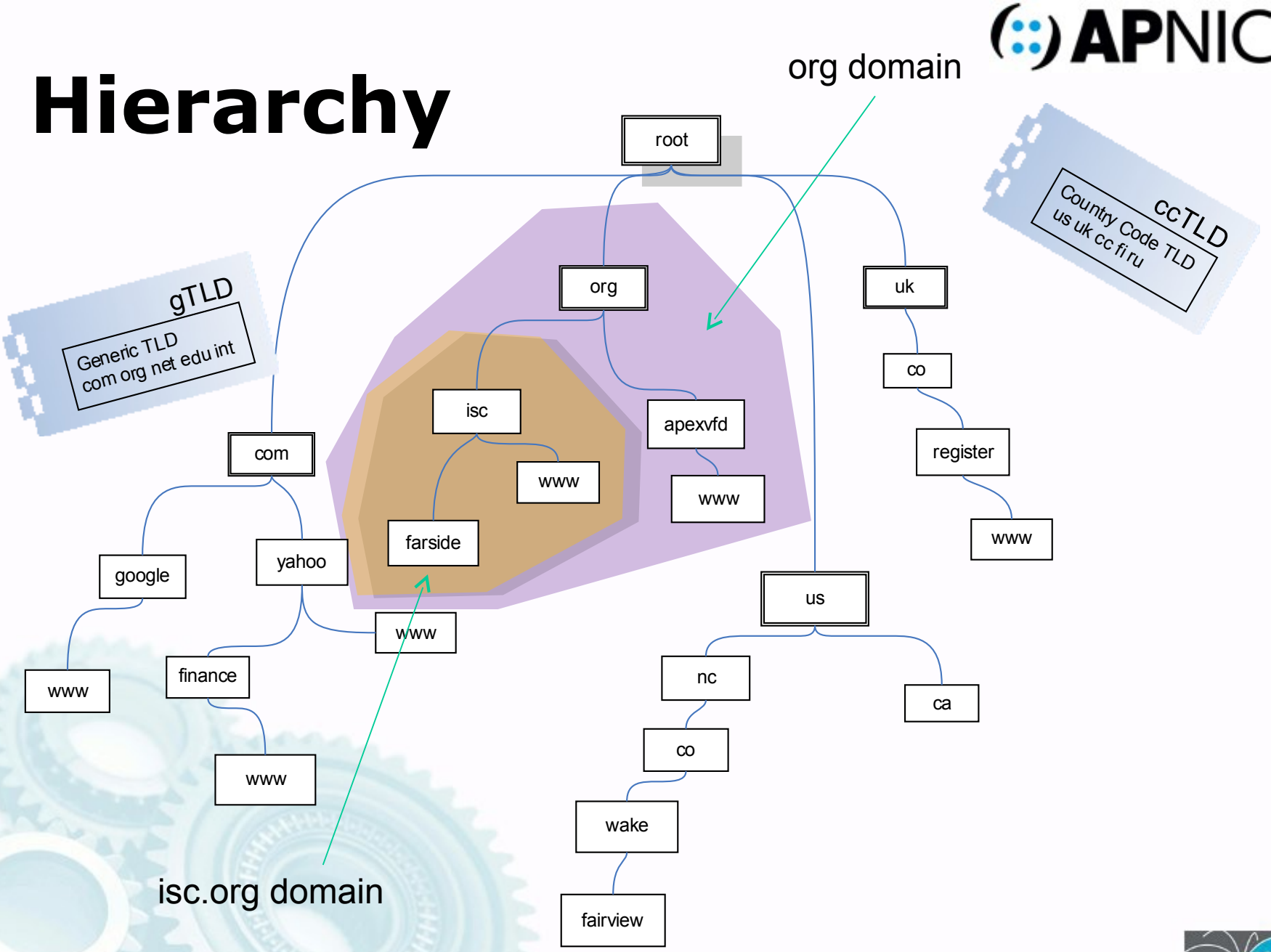

root
gTLD
Generic TLD
com org net edu int
ccTLD
Country Code TLD
us uk cc fi ru
com
org
uk
us
isc
apexvfd
co
register
www
www
www
farside
google
yahoo
www
www
finance
www
nc
ca
co
wake
fairview

Domains

- Domains are namespaces
- Everything below *.org* is in the *org* domain
- Everything below *isc.org* is in the *isc.org* domain and in the *org* domain



Hierarchy


org domain
root
ccTLD
Country Code TLD
us uk cc fi ru
gTLD
Generic TLD
com org net edu int
org
uk
co
register
www
isc
apexvfd
www
www
com
farside
yahoo
google
www
www
finance
www
www
us
nc
ca
co
wake
fairview
isc.org domain

Inverse DNS

- Mappings from an IP address to the associated name is in a different namespace

`in-addr.arpa.`

`ip6.arpa.`

Inverse DNS

- Similar to forward naming, inverse naming is from most specific to least specific data:

```
1.5.168.192.in-addr.arpa.
```

- Provides the name for the system at the IP address 192.168.5.1

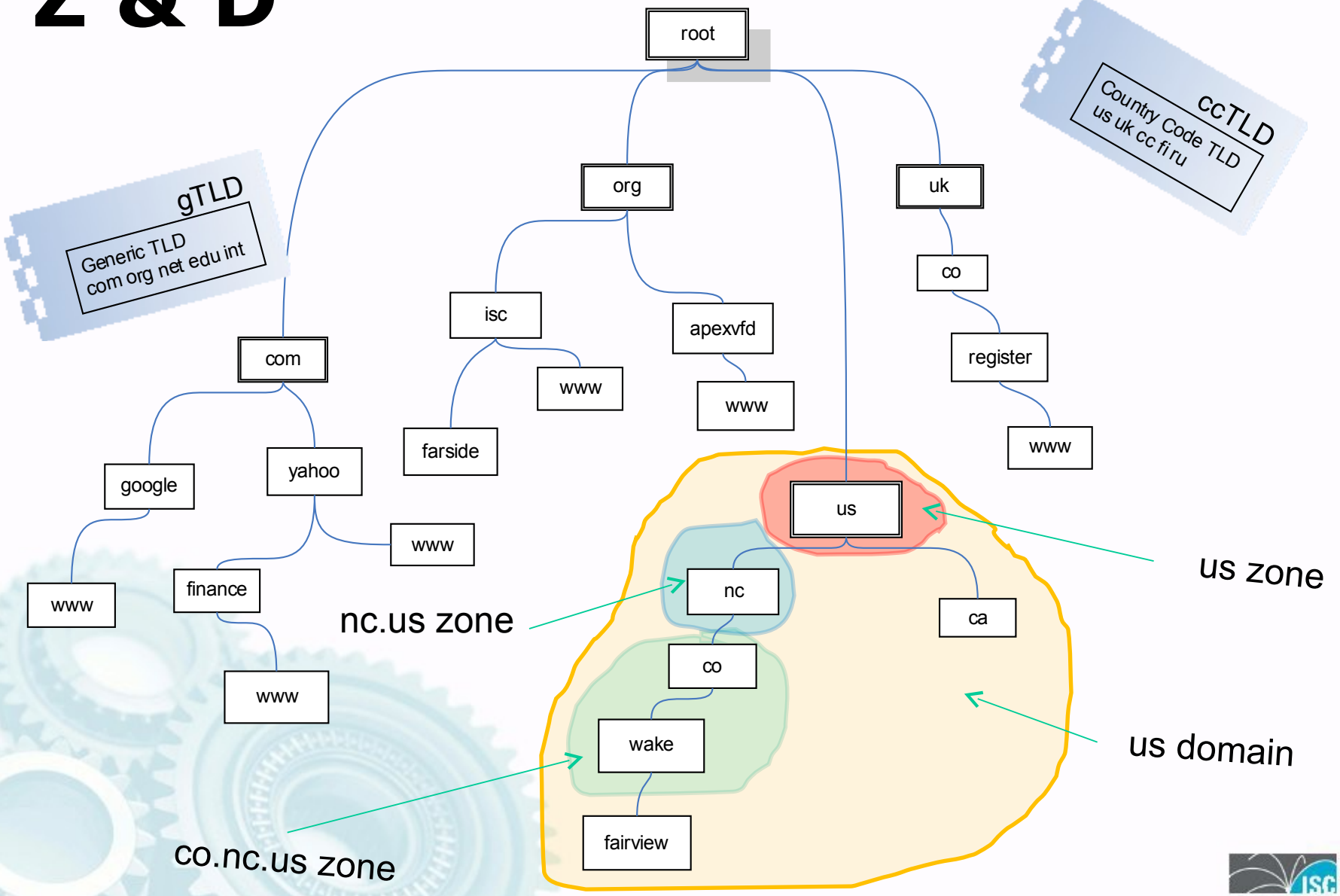
Delegation

- Administrators can create subdomains to group hosts
  - Geography, Organization, Politics
- Responsibility for the subdomain may be delegated to someone else
- Parent domain retains links to child

Zones and Domains

- Zones are administrative spaces
- Zone administrators are responsible for a portion of a domain's namespace
- Authority is delegated **from** a parent, and delegated **to** a child

Z & D


root
gTLD
Generic TLD
com org net edu int
ccTLD
Country Code TLD
us uk cc fi ru
org
uk
co
isc
apexvfd
register
com
www
www
www
farside
google
yahoo
www
us
www
finance
nc
ca
nc.us zone
us zone
co
www
wake
us domain
fairview
co.nc.us zone

Review -- Purpose of DNS

- Addresses are used to locate objects
- Names are easier to remember than numbers
- Humans want to access objects by name
- DNS fills this need

Understanding DNSSEC

Introduction

- With millions of recursive, caching servers on the Internet…
  - Each one needs to be able to be able to look up data from millions of zones
  - There is no way to distribute secret keys
    - Existing technology (TSIG) did not scale well

Introduction

- Central concept:

DNS data is augmented by a signature

- Validating resolvers can use the signature to verify that the data is authentic

Introduction

- DNSSEC is based on public key (asymmetrical) cryptography
  - Private key is used to sign DNS data
  - Public key is published via DNS so that validators can retrieve it
  - The public key is then used to validate the signatures, and there-by, the DNS data

Introduction

- DNSSEC provides cryptographic proof that the data received in response to a query is un-modified

- It does not deal with validating dynamic updates, nor with master to slave data transfers

Introduction

- DNSSEC enabled authoritative servers provide digital signatures across RRsets in addition to "standard" DNS responses

- DNSSEC validating resolvers provide authenticated responses with proven integrity

Introduction

- Clients using validating resolvers get "guaranteed good" results

- Data that does not validate provides a "**`SERVFAIL`**" response from the upstream resolver

Trust Validation

- With this knowledge, we are able to prove that data hasn't changed between the authoritative server and the validator, but how do we know we can trust it?

- Now that the root (".") is signed, that's easy, right?

Trust Validation

- DNSSEC is based on chains of trust
- At the top of chains are "trust-anchors"
  - One (signed) root, one trust-anchor
  - Until all TLDs are signed, it's not so easy
  - Trust anchors must be gathered and added to DNS configuration through leaps of faith

Trust Validation

- In BIND, trust anchors are added in "trusted-keys" statements

```
trusted-keys {
   . 257 3 8 "AwEAA[..]ihz0=";
};
```

- This creates an anchor based at the DNS root from which a chain is created

Chain of Trust

- Once a "trust anchor" is inserted, how does it actually create trust that leads down the DNS tree?

- Trust anchors consist of bits capable of validating the key used to sign the key that signs data in a given zone

Chain of Trust

- First, we must realize that there are TWO keys inserted into each zone
  - Zone Signing Key (ZSK)
    - Used to sign the resource records in the zone being secured
  - Key Signing Key (KSK)
    - Used to sign the Zone Signing Key

Chain of Trust

- Delegation of signed zones include a new Resource Record type
  - Delegation Signer – `DS`
  - Hash of the public portion of the child's Key Signing Key

Chain of Trust

- If the `DS` record in the parent is signed using the parent's zone signing key, we know that the `DS` record is valid.

- If the hash of the child's Key Signing Key record matches the `DS` record then we know that the Key Signing Key is valid.

Chain of Trust

- If the Key Signing Key is known to be valid, its signature of the Zone Signing Key proves that the Zone Signing Key is valid.

- If the Zone Signing Key is known to be valid, it can be used to validate other RRs in the zone.

Chain of Trust

- A living example:

`www.isc.org`

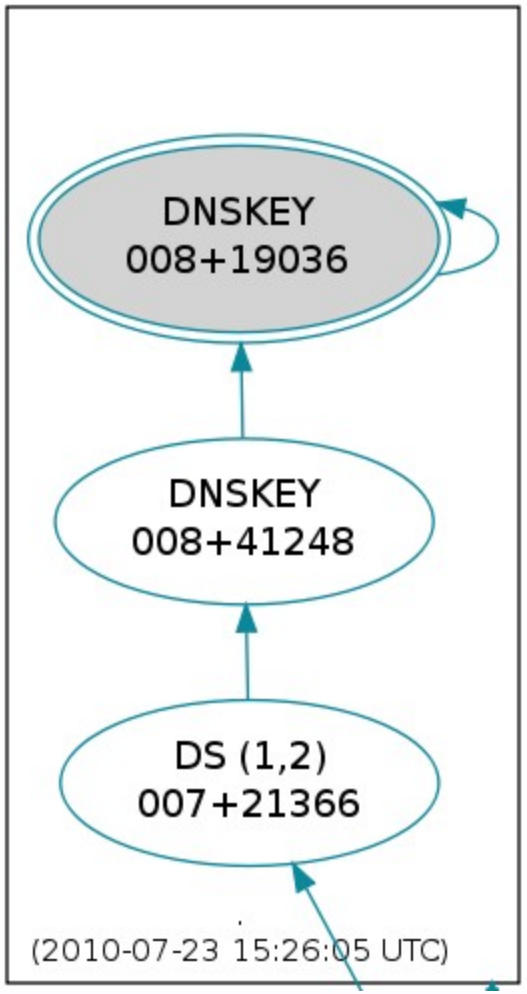
The following slides were created using Sandia National Laboratories "DNSViz"

http://dnsviz.net/

Trusting `isc.org`

. (root)

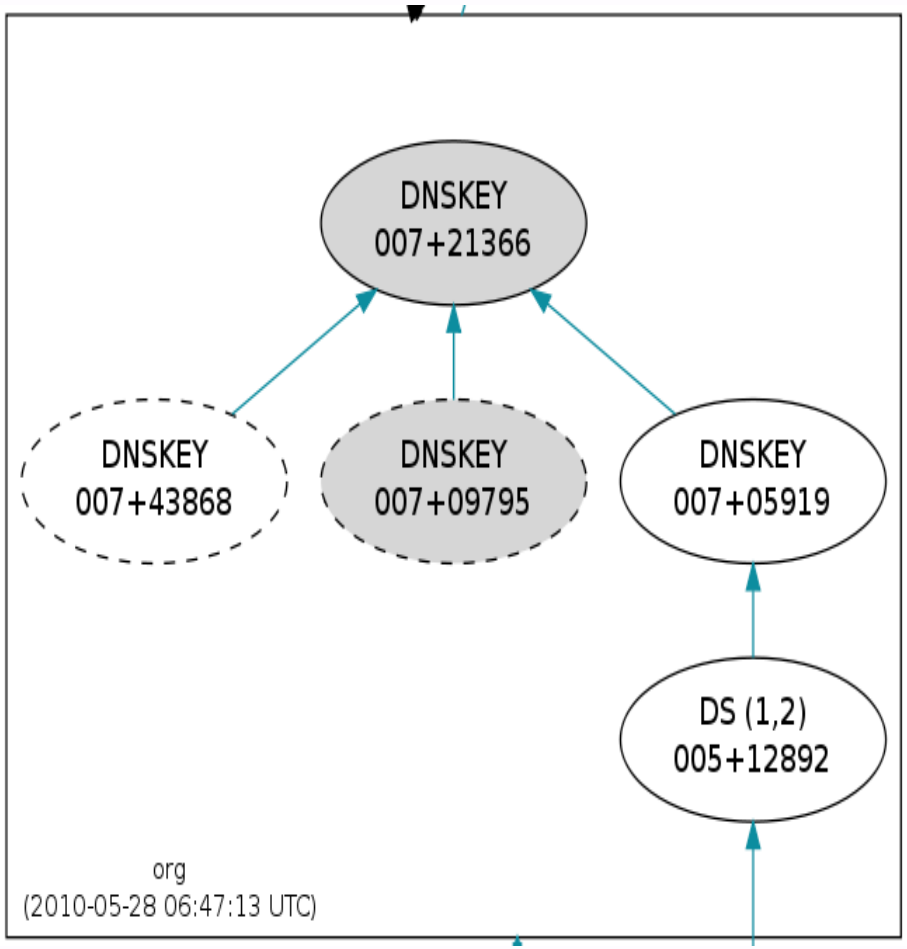
- KSK 19036
- ZSK 41248
  - Signed w/19036
- .org `DS` records
  - signed w/ 41248



Trusting `isc.org`

.org

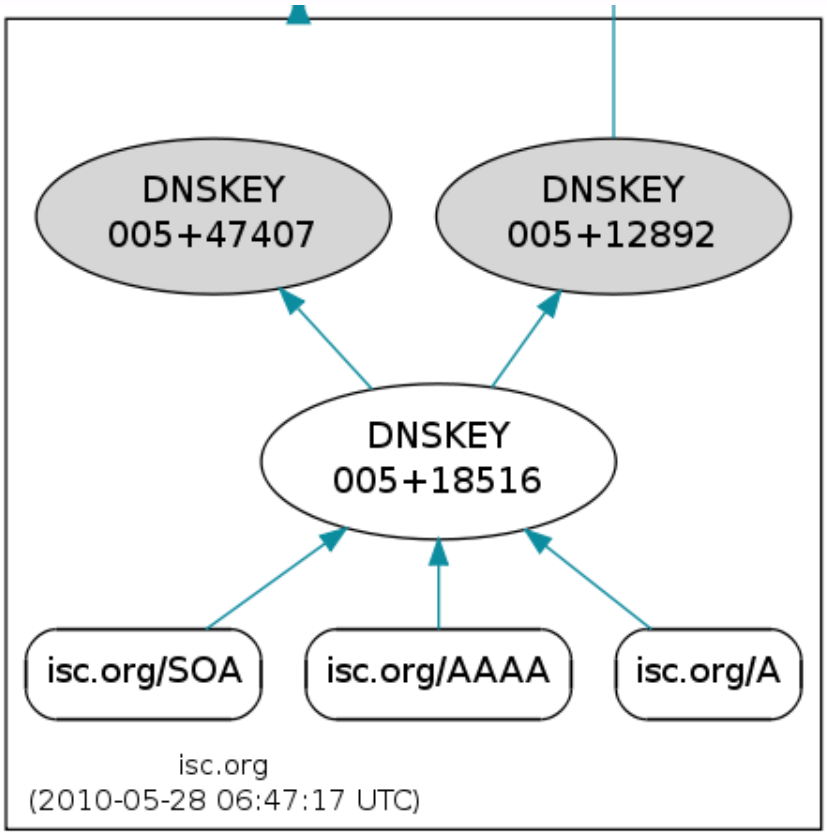
- KSK 21366
- ZSK 05919
  - Signed w/21366
- isc.org `DS` records
  - signed w/ 05919



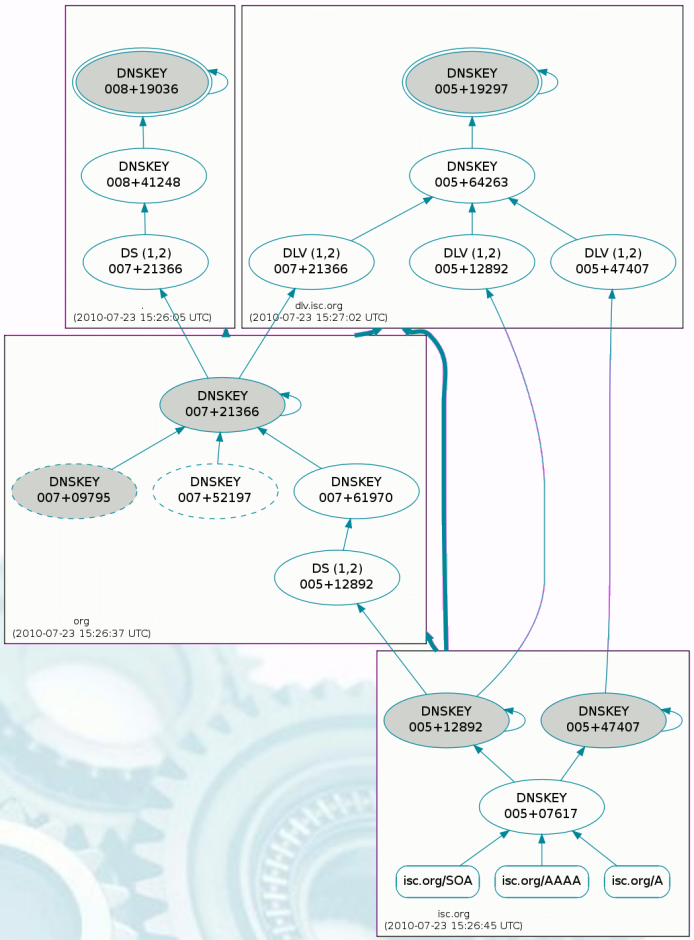
Trusting `isc.org`

isc.org

- KSK 12892
  - Hashed into DS
- ZSK 18516
  - Signed w/ 12892
- SOA, AAAA, A
  - Signed w/ 18516



Trusting `isc.org`



- With a trust anchor for root we can trust anything below it that is signed
  - And that has `DS` records in place

Why DNSSEC?

- 5 years ago, talking about DNSSEC being deployed anywhere would cause laughter.

- 3 years ago, talking about DNSSEC being deployed to the root would result in laughter.

Why DNSSEC?

- 2 years ago, Verisign and ICANN provided proposals as to how the root would be signed.

- On July 15, 2010, the first full production DNSSEC root zone was signed.

Why DNSSEC?

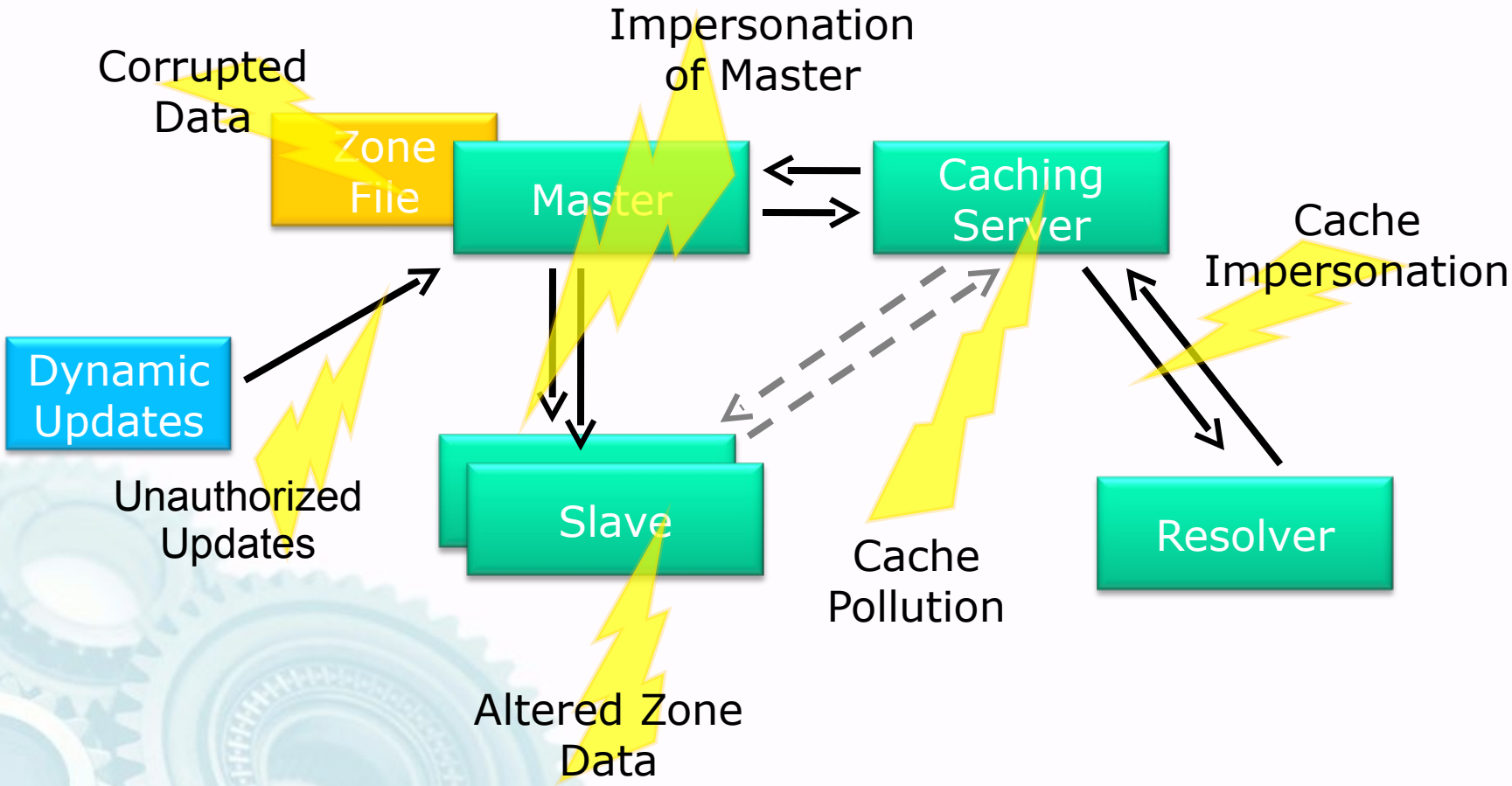
- What spurred this rapid change in attitudes?
- Was the world as we knew it about to end?
- Possibly.

Why DNSSEC?

- Contemplate for a moment the amount of trust that we put into the DNS infrastructure

- If DNS were to suddenly become unreliable or untrustworthy, what would the result be?

Why DNSSEC?


Corrupted Data
Zone File
Impersonation of Master
Master
Caching Server
Cache Impersonation
Dynamic Updates
Unauthorized Updates
Slave
Cache Pollution
Resolver
Altered Zone Data

Why DNSSEC?

- There are lots of places for "bad people" to do "bad things" in the DNS infrastructure.

- Aren't these the same problems that DNS has had since 1983?

- Yes, but...

Why DNSSEC?

- January 2008 – Life as normal
- February 2008 – Dan Kaminsky makes contact with ISC
- Since then, nothing has been normal...

Why DNSSEC?

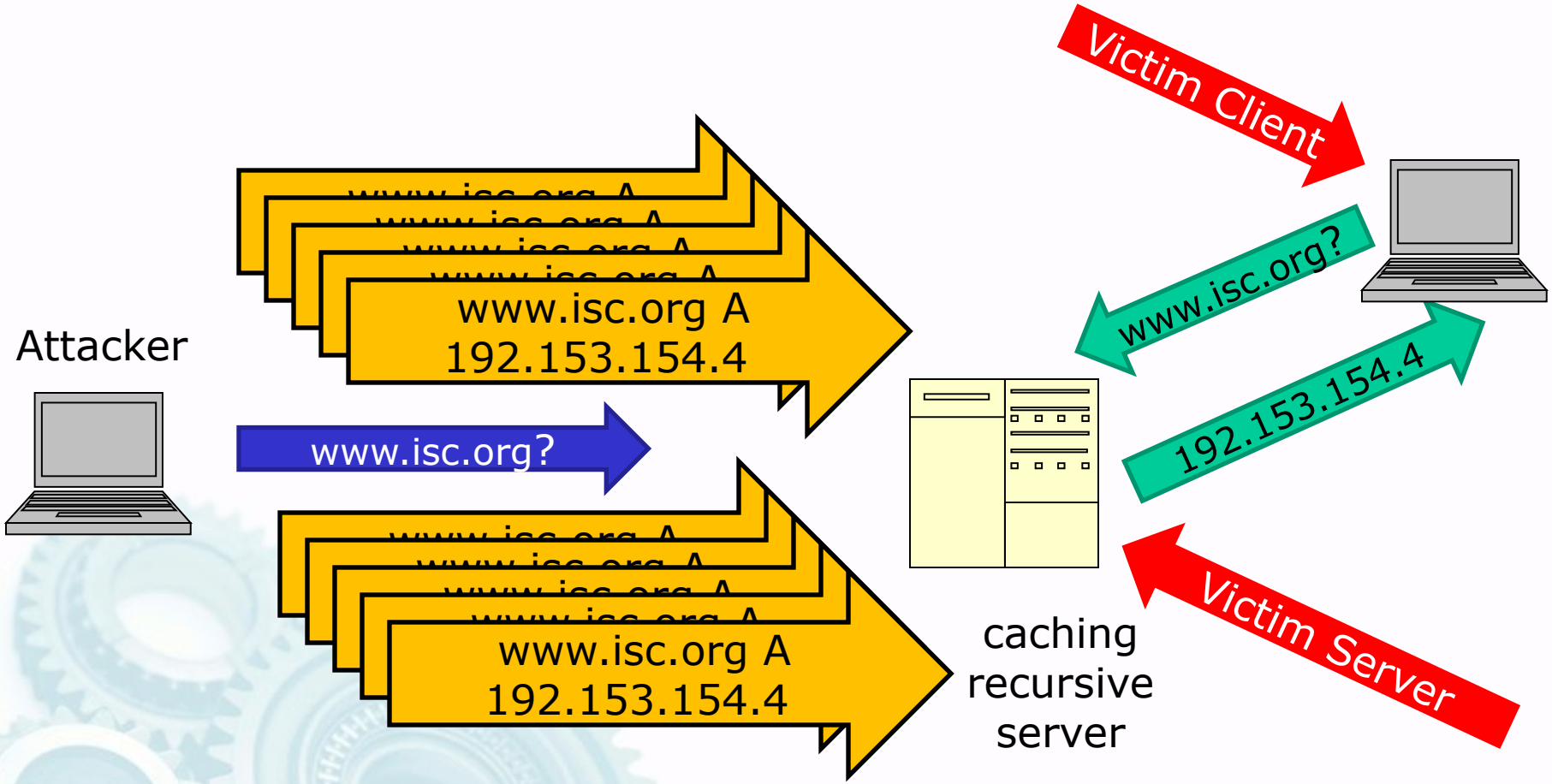
- In February, ISC, Microsoft, Cisco and other vendors were notified of the new DNS attack vector

- An effort was undertaken to provide software updates that would be released simultaneously across multiple platforms

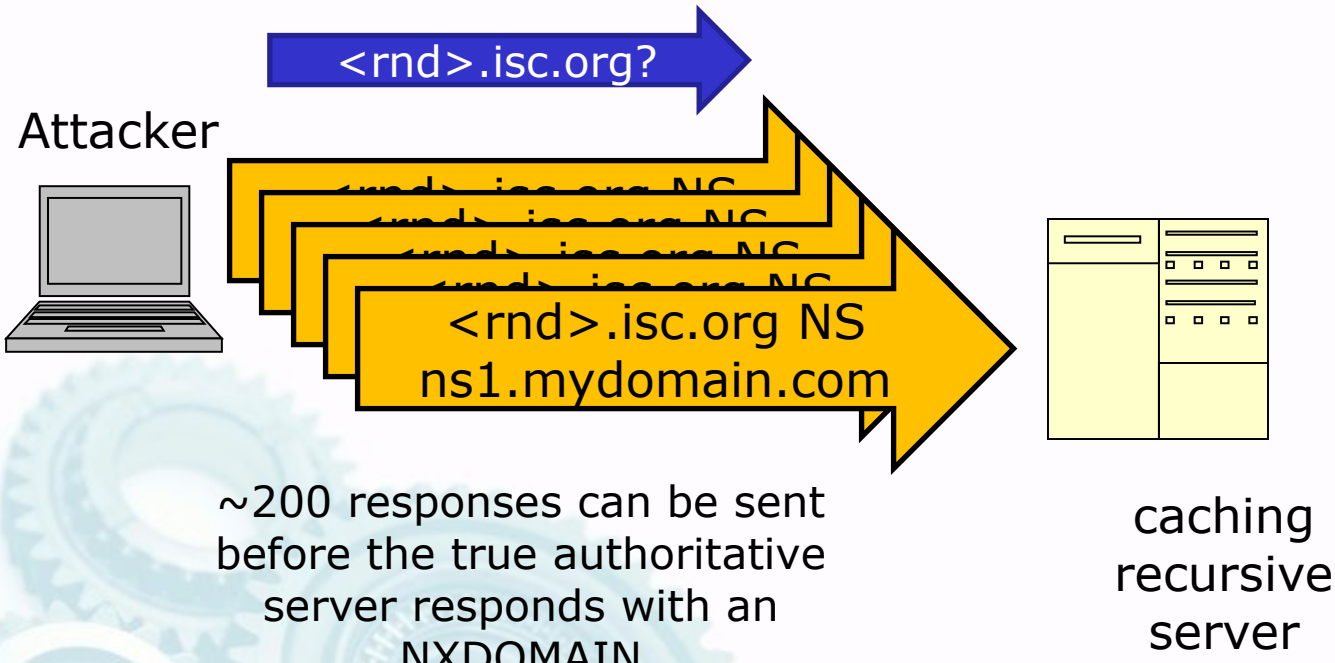
Why DNSSEC?

- "Classic" cache poisoning was known about for years.
- What was so magical about the "Kamisky" attack?
- It's all about timing... and scale...

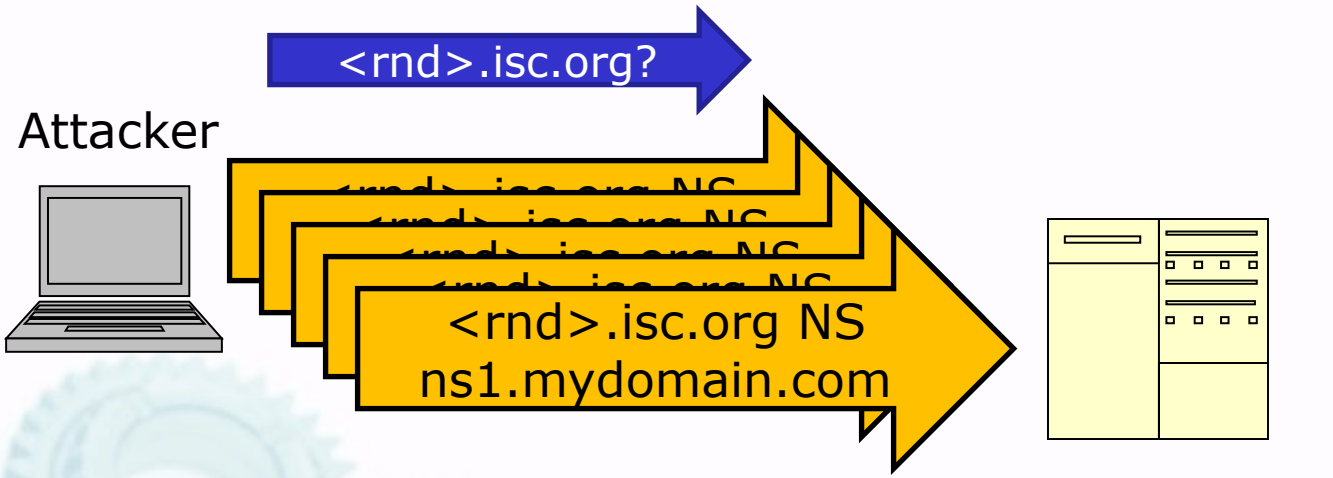
Classic Cache Poisoning



Kaminsky **NS Poisoning**



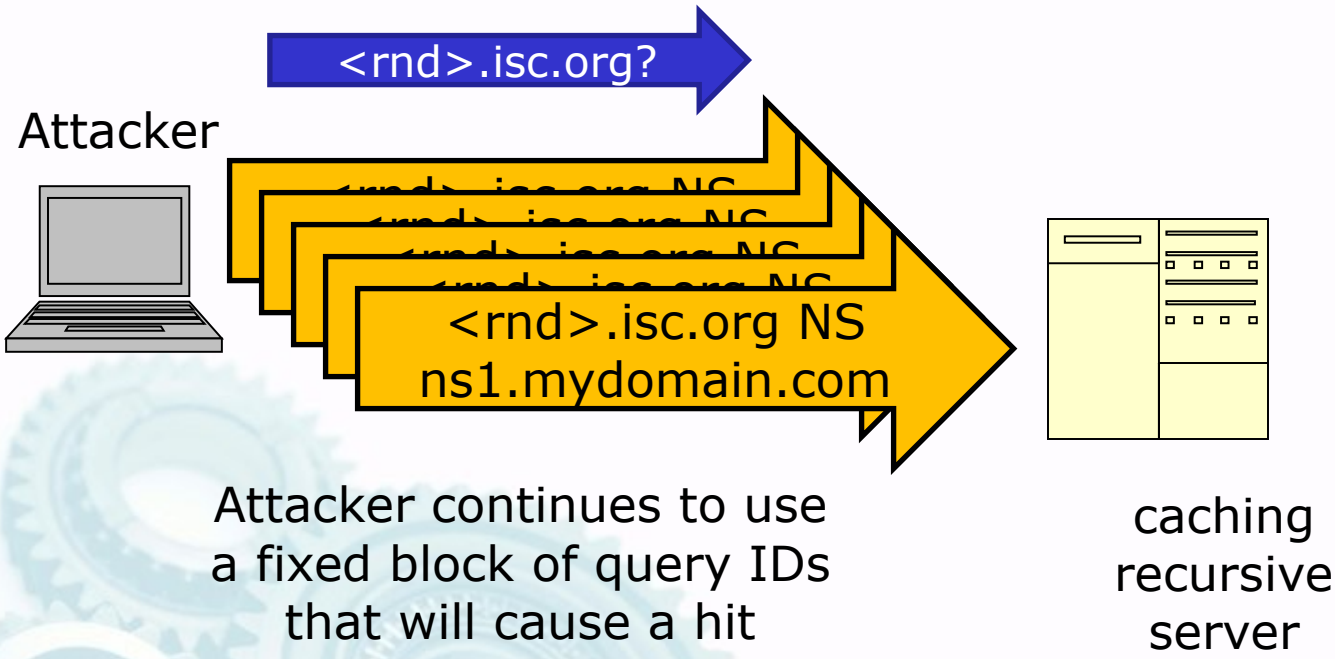
Kaminsky **NS Poisoning**



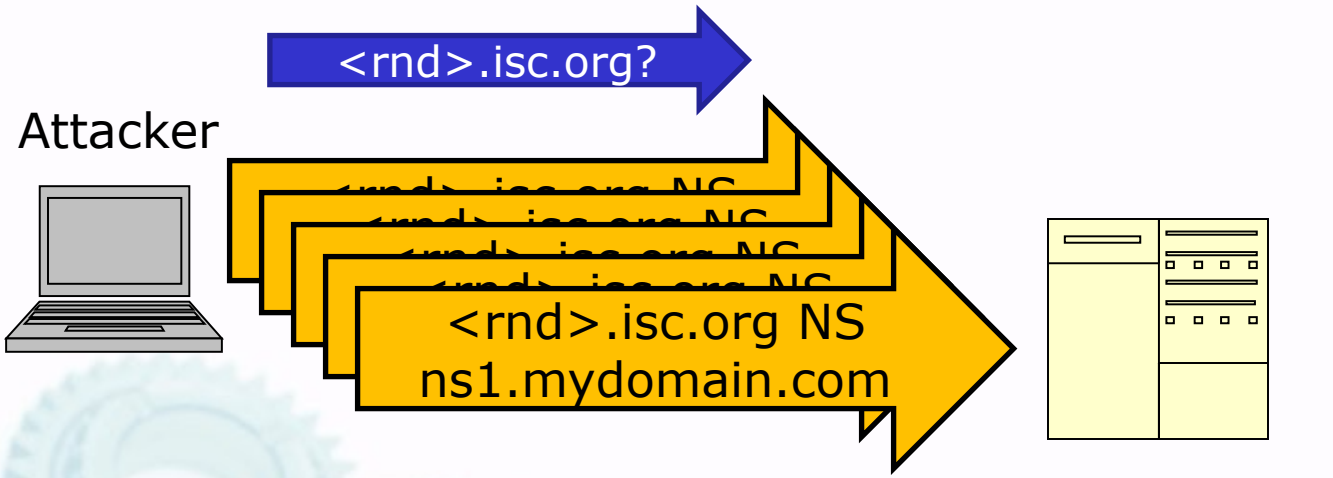
Since <rnd> is different at every query, the TTL of cached data is no-longer a deterrent

caching recursive server

Kaminsky **NS Poisoning**



Kaminsky **NS Poisoning**

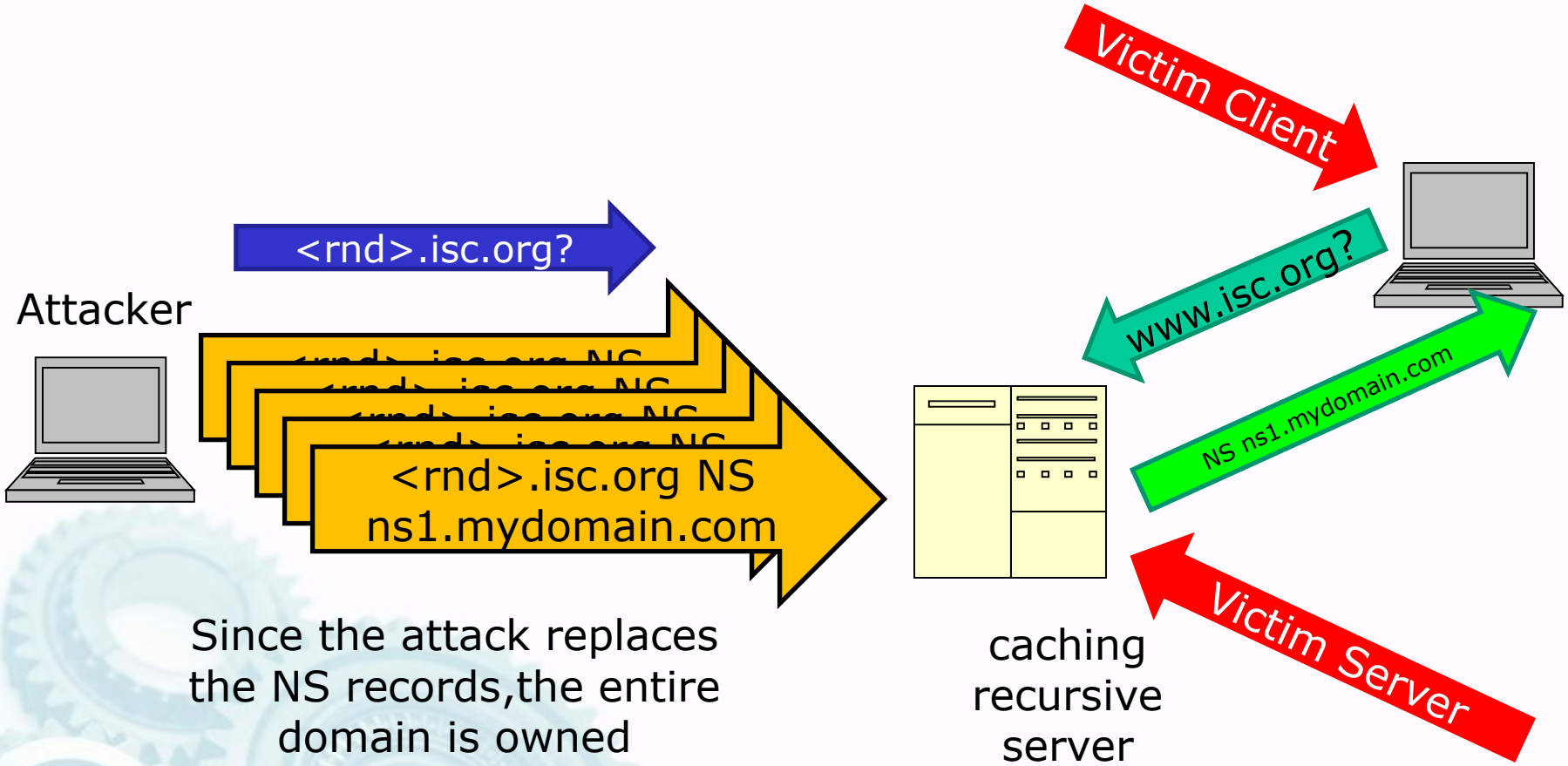


65536 / 200 = 327

Attacker has to send 327 <rnd> queries to guarantee a hit

caching recursive server

Kaminsky **NS Poisoning**



Why DNSSEC?

- This vulnerability was a "game-changer"
- Suddenly, caching servers across the Internet were vulnerable
- While a work-around (port randomization) was put into place, it is not a long-term fix

Why DNSSEC?

- In addition to the Kaminsky attack, what about untrusted environments?
- The hotel that you stay in as you travel
- The network that you attach to in the airport...

Why DNSSEC?

- If I run your recursive server, I can provide you with any response that I want..

```
www.google.com => 10.0.0.1
```

- Does the consumer believe that?

Why DNSSEC?

- DNSSEC provides the ability to validate responses to insure that the response is unaltered since it left the authoritative server.

- DNSSEC data can, if you wish, be ignored.

So…

Need Determination

- Who needs to deploy DNSSEC now?
  - Government Mandate
    - `.gov` directly
    - service providers to `.gov`
  - The "cool kids"

Need Determination

- Who should deploy DNSSEC now?
  - Those interested in protecting their customers
  - Future-proofing
    - Better to blow it now than in a year

Hardware Overview

- Does DNSSEC mandate the need for additional hardware?
  - Some changes may need to be made in architecture
  - Resource Records now have crypto-signatures which must be generated

Hardware Overview

- Authoritative Servers
  - Master server signs the zone
    - Off-line or possibly hidden master
  - Authoritative servers serve zone
    - Larger files
    - Larger responses
    - Higher network utilization

Hardware Overview

- Recursive Servers
  - Validation requires crypto-math
  - Larger responses
    - RRSIG records
    - In all sections of the response packet

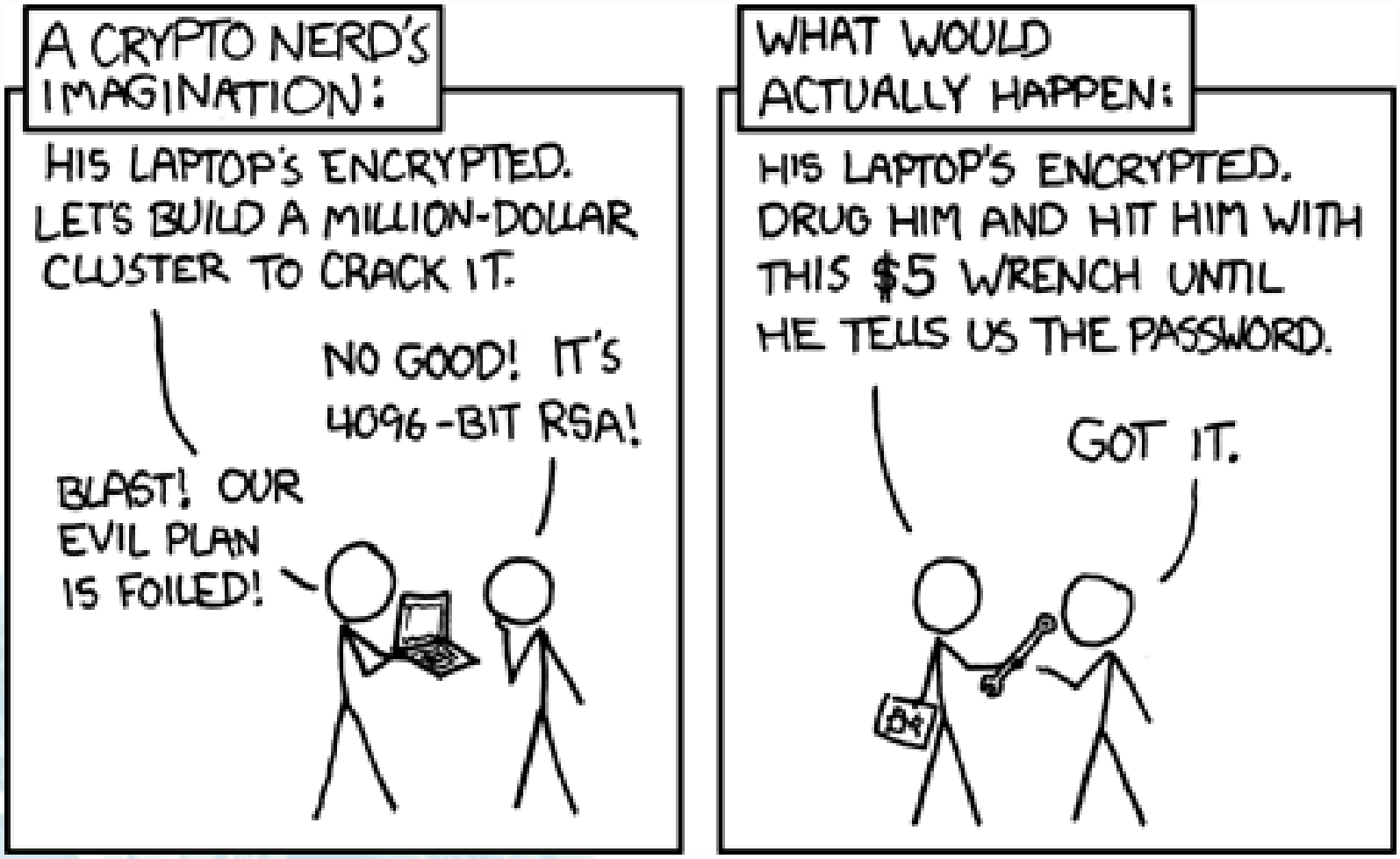
Crypto Acceleration

- Hardware speeds up signing and validation
  - Offload "hard math" from CPU
  - Increase speed of signing large zones
  - Increase speed of validating responses
  - Supported in BIND 9.6 and above

Securing the Keys

- You now have "secret" material in the filesystem...

- Do you trust everyone that has access to your system?



www.xkcd.com/538

Securing the Keys

- If your zone/data is worth physical injury to the employees
  - Consider a Hardware Security Module
- Without access to the keys, you'll either not be bothered or you'll be killed immediately

Securing the Keys

- ICANN root server signing platform
- For ccTLD's
  - PCH/ICANN DNSSEC signing platform
  - Build your own
  - Managed DNS Services (+politics)
- For Enterprises
  - Managed DNS Services
  - Build your own.

HSM

- Hardware Security Module
  - Keeps the keying material in a tamper-proof box
    - Intrusion
    - Temperature
    - Power fluctuation
  - Much more strict "startup"
    - Two "security officer" smartcards w/ PIN

HSM

- Private key portions are in the HSM
  - Public portion is in filesystem
- Data is sent into the HSM
  - Signed data returns

Processes

- How are zone changes handled currently?
  - Dynamic updates?
    - Instant Win!
  - Database insert/delete?
    - Add one step and Win!
  - Someone running 'vi'?
    - A bit of work to do here, but still workable

Extension of Processes

- Everyone will have changes in process
  - The size of the change depends on your current process

Signing Requirements

- Signing takes place on the authoritative server
  - On-line
    - Visible to the world
    - Hidden master
  - Off-line
    - Air-gap

NSEC vs NSEC3

- Proof of non-existance
- As defined, NSEC allows zone enumeration
  - NSEC3 dissuades "zone walking"
  - NSEC3 increases CPU usage on `NXDOMAIN` responses

Key Length

- Longer is always better, right?
  - Not so much.
  - Longer keys lead to longer signatures
    - Longer signatures lead to larger packets
      - Larger packets lead to more network traffic

Key Length

- NIST has recommendations:

  KSK – 2048 bit

  ZSK – 1024 bit

- These are defaults in BIND key generation

Key Rollover

- The longer a key is in the public view, the more vulnerable it is
- The longer in bits a key is, the more secure it is
- If a key is stolen, it doesn't matter how many bits are in it...

Key Rollover

- The method of rolling a key depends on the type of key being rolled

  KSK – Double Signature

  ZSK – Pre-Publication

- Timing is CRITICAL!

Testing the System

- Having deployed DNSSEC, how do you test it?
  - Deploy at least one validating recursive server
  - Watch for abnormalities

Quality Assurance

- Practice, Practice, Practice
  - Make sure that tools notice when something goes wrong
  - Signing process
  - Publication process
  - Key rolling

A Real DNSSEC Deployment
Portuguese TLD

Signing .pt

Existing Infrastructure

- Zones being added to .PT were inserted into a database
- 2x Daily runs provided zone data for the master server
- When verified, data was AXFRd to the slave servers

Recommended Changes

- Modify system to use dynamic zone on hidden master
- As database inserts are done, dynamic inserts into zone are also done
- Signed records are automatically generated by BIND

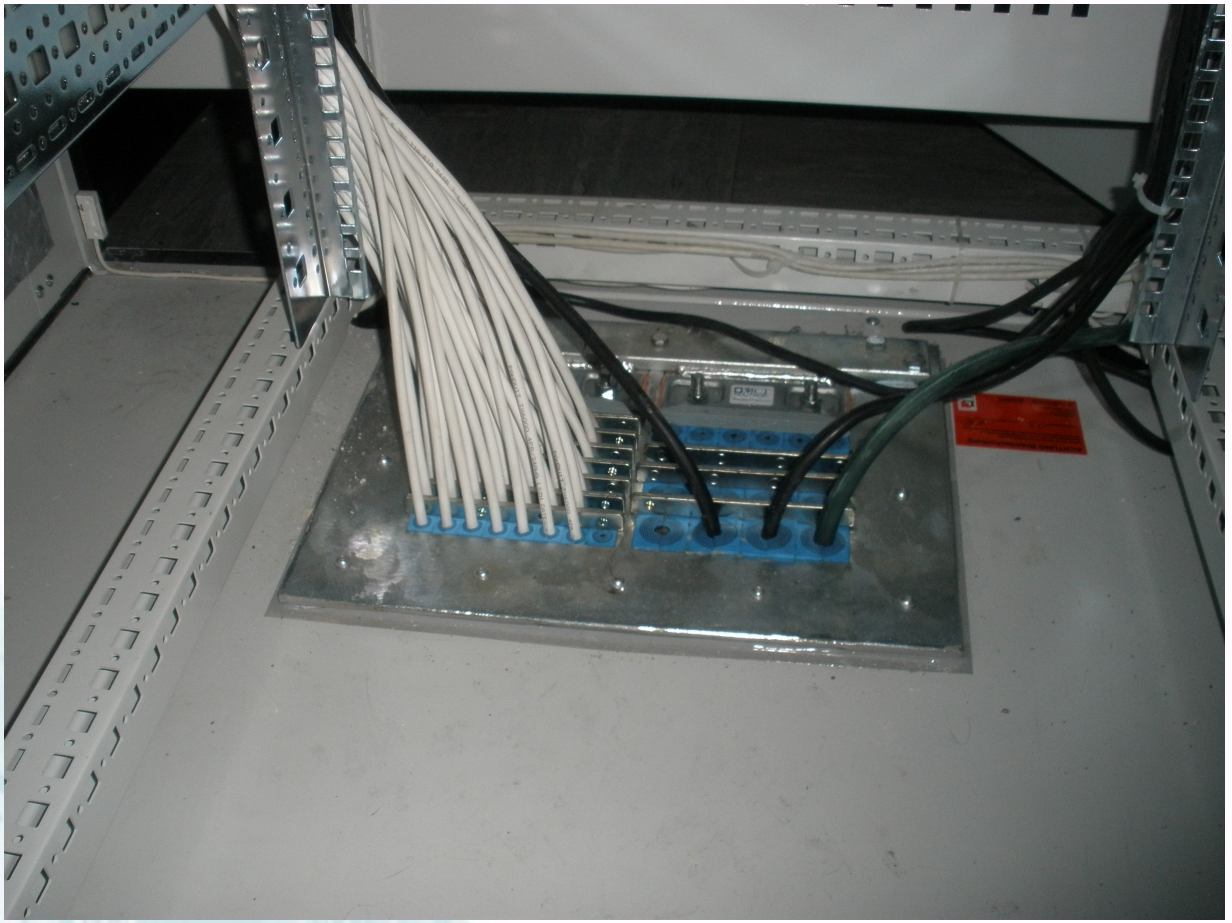

Império Bonança
97-FC-45

Lampertz "Vault"

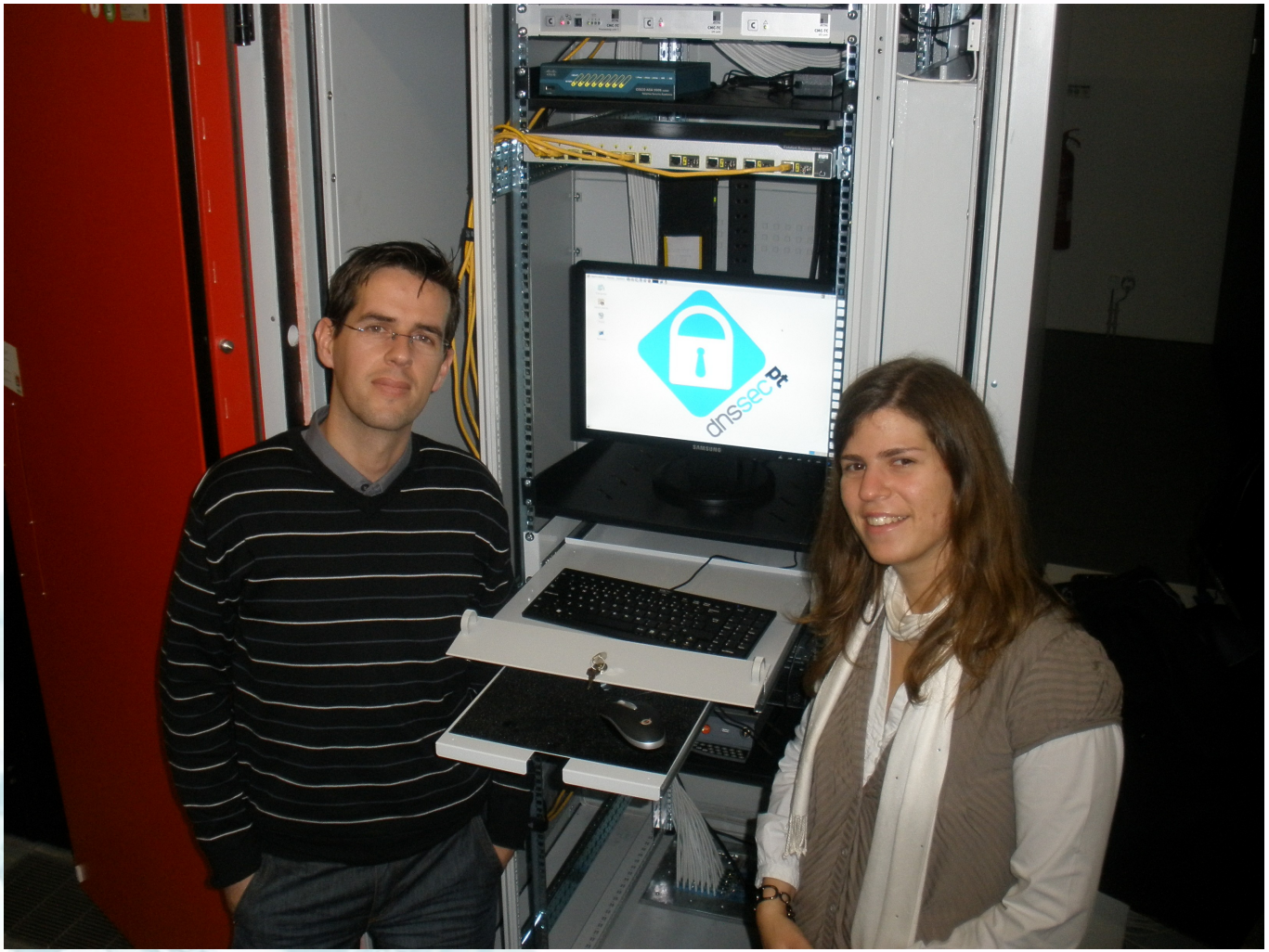
Self-cooling!

Wired Fire Alarm

External Connectivity



Assis and Sara



AEP Keyper


Boutique da Carne III
grupo victor victor
comércio de carnes
Boutique da Carne III
ENTRADA
LEITÃO ASSADO DE NEGRAIS C/MOLHO ESPECIAL
CABRITO INTEIRO OU METADE
12,98€
-PERNA PERÚ RECHEADA
-FRANGO RECHEADO
-ROLO FINGIDO
PERUAS DE 4/6 Kg
A BOUTIQUE DA CARNE III, DESEJA A TODOS OS CLIENTES E AMIGOS, UM FELIZ NATAL E PRÓSPERO ANO NOVO.
-QUEIJOS REGIONAIS
-ENCHIDOS
-CARNES
FARINHEIRA/LINGUIÇA
AMEIXAS
TÂMARAS
CASTANHAS
ANANÁS/BACON
12,98 €/kg
CABRITO 12,98
Rolo Carne Picada e Almôndegas 2,98
Entrecosto Mirão 2,98
Lombinho Porco 4,95

DNSSEC Deployment

BIND 9.7

Recursive Server

Trust Validation

- In BIND, trust anchors are added in "trusted-keys" statements

```
trusted-keys {
   . 257 3 8 "AwEAA[..]ihz0=";
};
```

- But, what happens when a "hard-configured" key changes?

RFC-5011 ready anchors

- Be ready for KSK roll-over:

```
managed-keys {
      "." initial-key  257 3 8
       "AwEAA[..]k1ihz0=";
   };
```

- Defines the initial key used as KSK for the given zone

RFC-5011 ready anchors

- A file is created that tracks key changes

```
managed-keys.bind
managed-keys.bind.jnl
```

- This file will contain the currently active key, even if the configured key has rolled

RFC-5011 ready anchors

- Newly added "`rndc secroots`"
  - Creates a file "`named.secroots`" containing a list of the current managed keys that are in use:

```
10-Sep-2010 12:56:08.950

 Start view _default

./RSASHA256/19036 ; managed
dlv.isc.org/RSASHA1/19297 ; managed
```

RFC-5011 ready anchors

- One problem with `managed-keys`:
  - If a key has rolled without being noticed, validation will fail
  - This can happen if a validating server is off-line during a key roll-over, etc.

Authoritative Server

DNSSEC Deployment

- Generate required keys
  - `dnssec-keygen`
- Insert them into the zone
  - manual (or dynamic)
- Sign zone data
  - `dnssec-signzone` (or dynamic)
- Perform scheduled zone maintenance
  - manual (or dynamic)

DNSSEC Deployment

- `dnssec-keygen`
  - Used to create the required keys
    - Key Signing Key
    - Zone Signing Key

DNSSEC Deployment

- `dnssec-keygen`
  - Defaults algorithm to `RSASHA1`
  - Provides defaults for key size if default algorithm is used:
    - KSK – 2048 bits
    - ZSK – 1024 bits

DNSSEC Deployment

- `dnssec-keygen <zonename>`
- `dnssec-keygen -f KSK <zonename>`

- Produces 2 files per key

`K<zonename>+XXX+YYYY.key`

`K<zonename>+XXX+YYYY.private`

DNSSEC Deployment

- `dnssec-keygen`
  - Once keys are created, include their public portions (`.key`) into the zone file using standard procedures
  - Keep the `.private` portions secure

DNSSEC Deployment

- `dnssec-signzone`
  - Signs the zone data
    - Creates RRSIG resource records for each authoritative RRset in the zone
    - Transforms zone into "machine generated" file with a `.signed` extension

DNSSEC Deployment

- `dnssec-signzone`
  - BIND 9.7 introduced a new feature..
    - Smart Signing
      - Looks in key repository (directory) for keys
      - Keys are included in zone automatically
      - If key files contain timing meta-data, that timing data is used

DNSSEC Deployment

- `named`
  - New dynamic zone configuration
    - `update-policy local;`
      - Automatically creates "local-only" TSIG key
    - Allows BIND to update without complex configuration

DNSSEC Deployment

- `named`
  - New zone options for dynamic zones
    - `auto-dnssec off;`
      - Default
    - `auto-dnssec allow;`
      - Enables auto-inclusion of keys from repository
      - Enables "`rndc sign`"
    - `auto-dnssec maintain;`
      - Update DNSSEC based on key meta-data

DNSSEC Deployment

- `nsupdate`
  - New option `-l` (ell)
    - Use the named created "local key"
    - Set the `server` address to `localhost`

DNSSEC Deployment

- `rndc`
  - New option `sign`
    - Takes a dynamic zone, searches for keys in the key repository and signs the zone as needed.

Making it work...

```
zone test.com {
   type master;
   key-directory "keys";
   update-policy local;
   auto-dnssec maintain;
   file "dynamic/test.com.zone";
};
```

Making it work...

```
dnssec-keygen -K /etc/namedb/keys \
          test.com
dnssec-keygen -f KSK -K /etc/namedb/keys \
          test.com
rndc sign test.com
```

Zone is now signed and published

Zone will be automatically re-signed as needed

DNSSEC "just works"

- Adding or removing zone contents is now as simple as:

```
nsupdate -l
> update add <RRset>
> send
```

- `RRset` is added and signed data updated automatically

Timing Meta-Data

- `dnssec-keygen` creates meta-data in the key file:

`-P`	– Publication Date	(default: now)
`-A`	– Activation Date	(now)
`-R`	– Revocation Date	(none)
`-I`	– retIrement Date	(none)
`-D`	– Deletion Date	(none)

Timing Meta-Data

- These dates are used by `named` to maintain the zone signatures

- Date formats:

  `none` (literal)

  `YYYYMMDD`

  `YYYYMMDDHHMMSS`

  `now+<offset>`

  `y, mo, w, d, h, mi`

Timing Meta-Data

- To pre-publish a KSK without signing:

```
dnssec-keygen -K keydir \
  -f ksk -A none test.com
[...]Ktest.com.+005+11353

rndc sign test.com
```

Timing Meta-Data

- Once you are ready to sign the zone with the given key:

```
dnssec-settime -K keydir \
  -A now Ktest.com.+005+11353
rndc sign test.com
```

Timing Meta-Data

- To no-longer sign with the key, but leave it in the zone:

```
dnssec-settime -K keydir \
  -I now Ktest.com.+005+11353
rndc sign test.com
```

Timing Meta-Data

- And finally, remove the key from the zone:

```
dnssec-settime -K keydir \
  -D now Ktest.com.+005+11353
rndc sign test.com
```

Automation Warning!

Be aware that this automation does NOT deal with `DS` records in the parent or `DLV` records in a registry

DNSSEC Deployment

- BIND 9.7.2

`allow-new-zones` option

- boolean allowing creation of zones "on the fly"

`rndc addzone` / `rndc delzone`

- add and remove zones without manually editing `named.conf`

Create & Sign a zone

```
#!/bin/bash
cd /etc/namedb
cp template master/${1}

rndc addzone ${1} { type master\;\
                                file \"master/${1}\"\;\
                                update-policy local\; \
                                auto-dnssec maintain\; \
                                }\;

dnssec-keygen -f KSK -K /etc/namedb/keys $1
dnssec-dsfromkey -2 /etc/namedb/keys/K${1}.*.key > ds/${1}
dnssec-keygen -K /etc/namedb/keys $1
rndc sign ${1}
```

Create & Sign (NSEC3)

```
#!/bin/bash
SALT=`printf %04x%04x $RANDOM $RANDOM`
cd /etc/namedb
cp template master/${1}

rndc addzone ${1} { [..] };

nsupdate -l << //EOF
update add ${1} 30 IN NSEC3PARAM 1 0 10 $SALT

//EOF
dnssec-keygen -3 -f KSK -K /etc/namedb/keys $1
dnssec-dsfromkey -2 /etc/namedb/keys/K${1}.*.key > ds/${1}
dnssec-keygen -3 -K /etc/namedb/keys $1
rndc sign ${1}
```

DNSSEC Status in APNIC Region

- TLD's
  - Signed - .asia, .in, .jp, .lk, .mn, .my, .nu, .th
  - .{com,net,org} signed.
  - Testing - .kr, .nz, .hk
- APNIC in-addr.arpa/ip6.arpa zones
  - myAPNIC support for DS records.

APNIC's own DNSSEC Status

- APNIC in-addr.arpa/ip6.arpa zones.
  - All APNIC zones are now signed with NSEC.
  - myAPNIC now supports DS records (add/delete/maintain).

Questions or Comments?