DNSSEC

Introducción
Principios Generales
Despliegue

Vista General

Qué vamos a ver

- Los problemas que DNSSEC resuelve
- El protocolo y sus implementaciones
- Pautas para desplegar DNSSEC
- Problemas prácticos relativos al despliegue

Vista General

- El plan es demostrar un firmado de zona y tendremos instrucciones y herramientas disponibles de manera que usted pueda seguir por su cuenta si tiene una laptop con SSH (descargue el cliente si usa Windows)

Contenido

- Alcance del problema
- Repaso de DNS
- Conceptos básicos de DNSSEC
- Despliegue y operaciones
- Problemas pendientes y otros aspectos
- Status de DNSSEC hoy
- Demostración en vivo

Cuál es el problema?

Qué riesgos ?

- Vea la presentación de Dan Kaminsky para más detalles sobre los riesgos
  - Muchos escenarios
  - ➔ Secuestro de MX
  - ➔ Redirección de dominios completos
  - ➔ Sacar de servicio a un .COM grande
  - ➔ Falsificación completa de un banco en línea
  - ➔ Muchas más cosas divertidas
- Una guía ilustrada muy buena
  http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

Repaso

Recordemos

- El formato de zona de BIND es muy común, así que lo usaremos aquí:

```
zone. SOA nsX.zone. hostmaster.zone.
                  ( 2009022401  ; serial
                  1d            ; refresh
                  12h           ; retry
                  1w            ; expire
                  1h )          ; neg. TTL

zone.       NS  ns.zone.
            NS  ns.otherzone.

zone.       MX  5 server.otherzone.
www.zone.    A  1.2.3.4
...
```

Recordemos

- Estructura de los récords:

```
NAME          [TTL]  TYPE  DATA (type specific)
-------------------------------------------------
host.zone.     3600  A     10.20.30.40
sub.zone.     86400  MX    5 server.otherzone.
```

Recordemos

- Múltiples records con *el mismo nombre y tipo* se agrupan en Resource Record Sets (RRsets):

```
mail.zone.      MX      5 server1.zone.
mail.zone.      MX      10 server2.zone.
```
RRset

```
server1.zone.   A       10.20.30.40
server1.zone.   A       10.20.30.41
server1.zone.   A       10.20.30.42
```
RRset

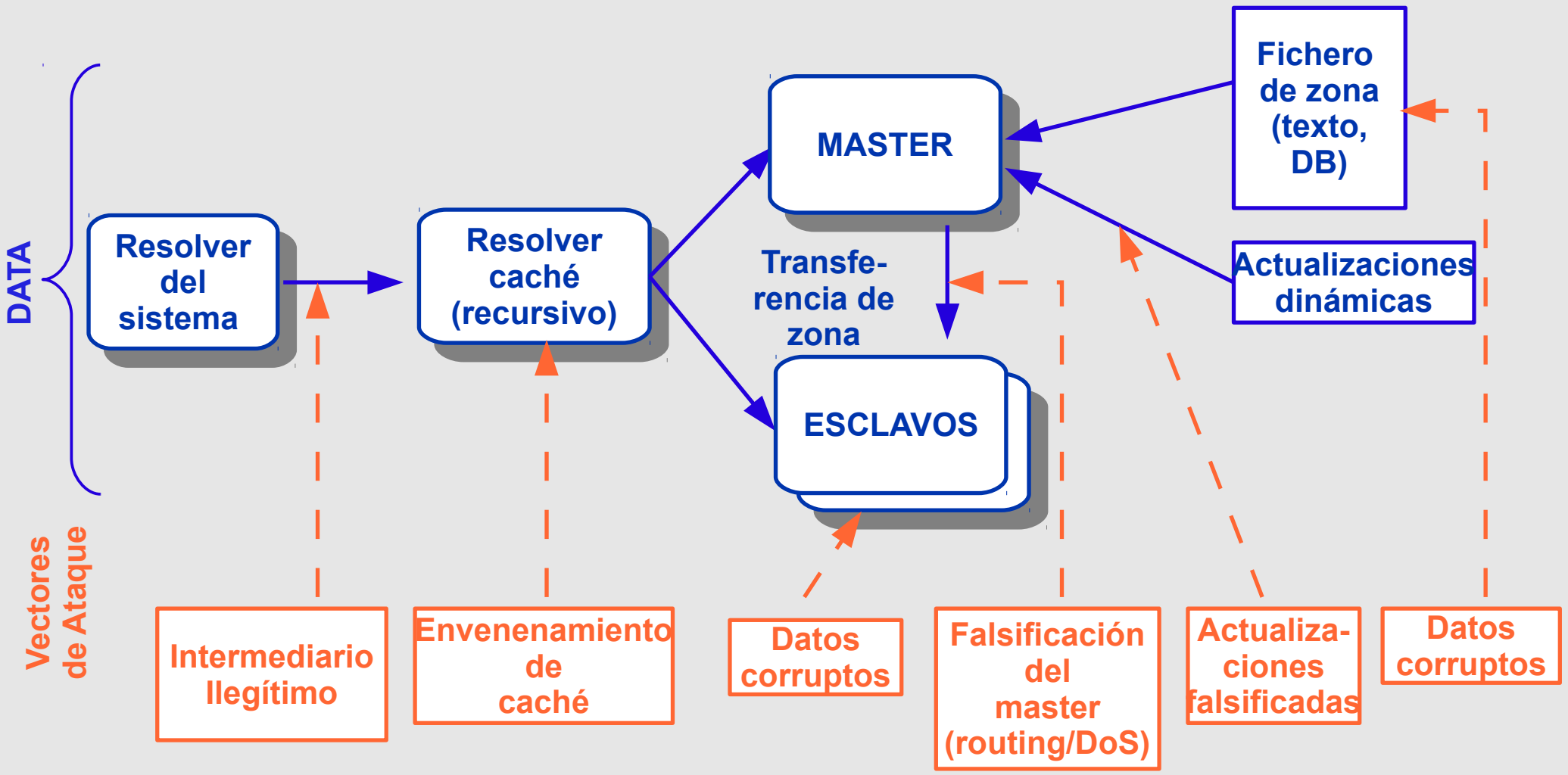
```
server1.zone.   AAAA    2001:123:456::1
server1.zone.   AAAA    2001:123:456::2
```
RRset

```
server2.zone.   A       11.22.33.44
```
RRset

Puntos de ataque de DNS

Flujo de datos de DNS

Puntos de ataque



Conceptos de DNSSEC

Repaso de criptografía de clave pública

DNSSEC – breve resumen

- Autenticidad de datos e integridad al firmar los RRSets con una clave privada
- Claves públicas (DNSKEYS), para verificar las firmas (RRSIGs)
- Los sub-dominios firman sus zonas con su llave privada
  - La autenticidad de la llave se establece gracias a la firma/checksum del record delegation signer (DS) por la zona superior
- Repetir en la zona superior...
- No es tan difícil en papel
  - Operativamente, es un poco más complicado

DNSSEC

DNS SECurity extensions

- Conceptos
- Nuevos récords (DNSKEY, RRSIG, NSEC/NSEC3 y DS)
- Nuevas opciones de protocolo (CD, AD, DO)
- Crear una zona segura
- Delegar la autoridad de firmado (signing authority)
- Renovación de llaves

DNSSEC - Conceptos

- Cambia el modelo de confianza de DNS de “abierto” y ”de confianza” a uno de ”verificable”
- Uso extensivo de criptografía asimétrica para lograr:
  - Autenticación del origen
  - Integridad de los datos
  - Autenticidad de la negación de existencia
- No se trata de proveer confidencialidad
- DNSSEC no implica una carga computacional en los servidores autorizados (!= los *firmantes*)
- No se cambia esencialmente el protocolo
  - Puede coexistir con la infraestructura actual
  - ➔ ... Bueno, más o menos (EDNS0)

DNSSEC - conceptos

- Crear una cadena de confianza usando el modelo existente basado en delegaciones para la distribución que es el DNS
- No se firma la zona completa, se firma un RRset

- Nota: la zona superior NO firma la zona inferior.
  La superior firma un *apuntador* (hash) a la *clave* usada para firmar los datos en la zona inferior (importante!)

Nuevos Records

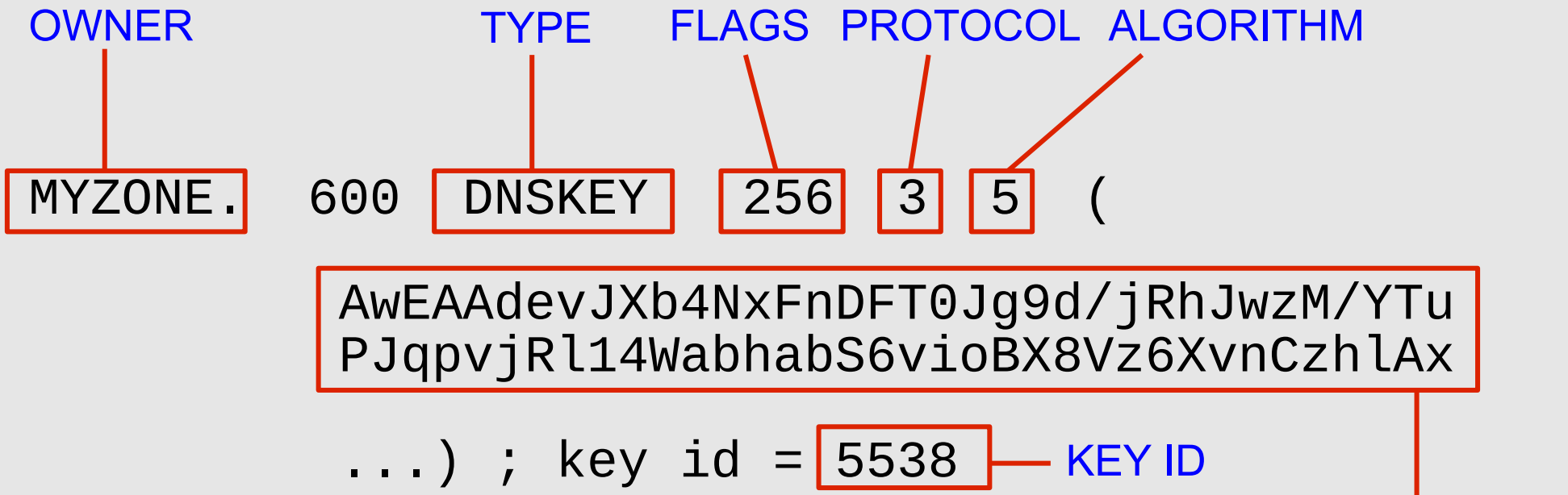
DNSSEC: nuevos RRs

Se añaden cuatro nuevos Resource Records*:

1. **DNSKEY**: Llave pública utilizada en el proceso de firmado.
2. **RRSIG**: Firma de un RRset
3. **NSEC/NSEC3**: Provisto como evidencia de que el nombre y/o el tipo de RR no existe
4. **DS**: Delegation Signer. Contiene el *hash* de la clave pública usada para firmar la llave que a su vez servirá para firmar los datos de la zona. Se siguen los RRs DS hasta encontrar una zona “de confianza” (idealmente la raíz).

***Vea la excelente discusión de Geoff Huston en**
http://ispcolumn.isoc.org/2006-08/dnssec.html

DNSSEC: DNSKEY



- FLAGS determina el uso de la llave (más sobre esto...)
- PROTOCOL siempre es 3 en la versión actual de DNSSEC
- ALGORITHM puede ser:
  - 0 – reserved
  - 1 – RSA/MD5 (deprecated)
  - 2 – Diffie/Hellman
  - 3 – DSA/SHA-1 (optional)
  - 4 – reserved
  - 5 – RSA/SHA-1 (mandatory)
  - 8 – RSA/SHA-256

http://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xml

DNSSEC: DNSKEY

- En la práctica hay dos pares de claves DNSKEY por cada zona:
  En un principio, un par de claves (pública, privada) definido por zona:
  - privada para firmar los datos de la zona (RRsets)
  - pública publicada en el record DNSKEY de la zona
  - Récord DS (DNSKEY hash) publicado en la zona superior, y firmada a su vez con el resto de los datos
- Problema con usar una sola clave:
  Para renovar esta clave, debe actualizarse el record DS en la zona superior (ya que el DS es una huella de la clave pública)
  - Se introduce la Key Signing Key (flags = 257)

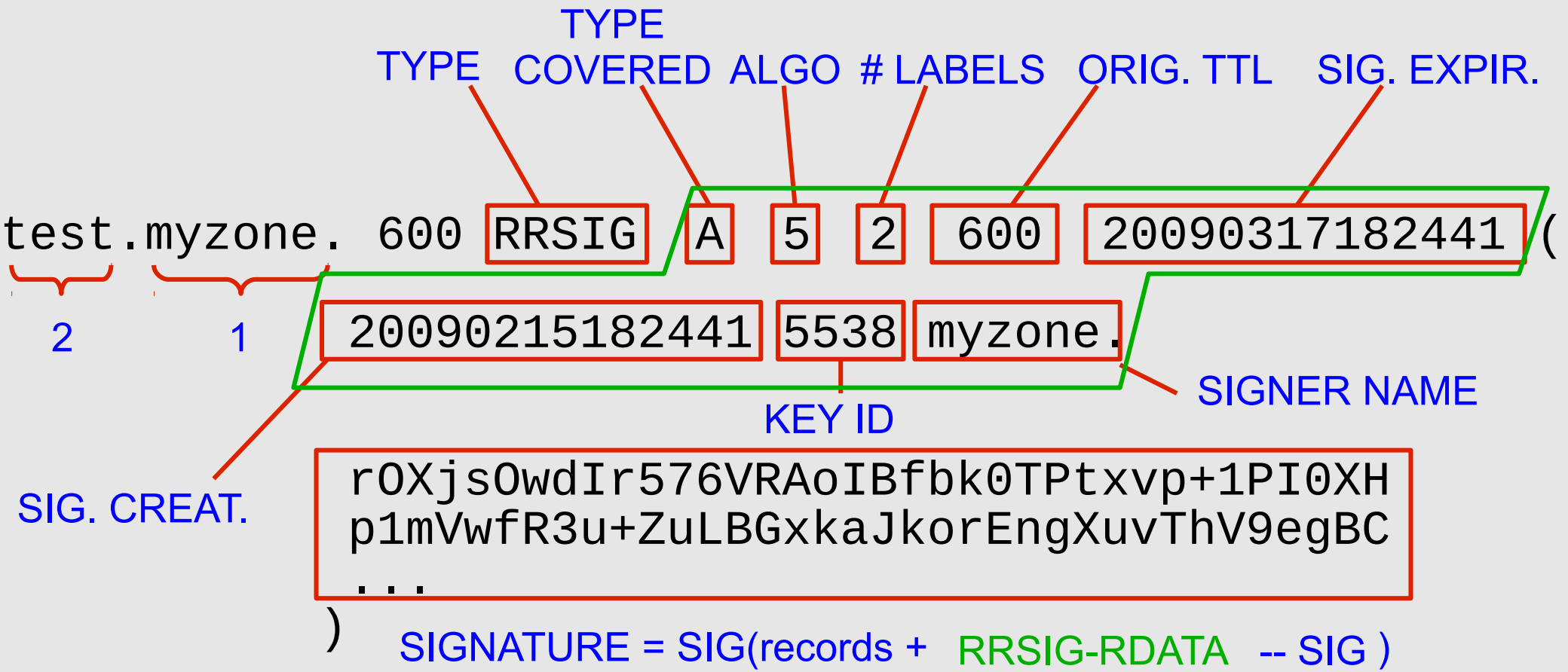
DNSSEC: KSK y ZSK

- Para permitir la renovación de claves (“rollovers”), se generan dos pares de claves:
  Key Signing Key (KSK)
  - Referenciada por la zona superior (Secure Entry Point), en forma de DS (Delegation Signer)
  - Usada para firmar la Zone Signing Key (ZSK)

  Zone Signing Key (ZSK)
  - Firmada por la Key Signing Key
  - Usada para firmar los RRsets
- Esta disociación permite la renovación del ZSK sin tener que renovar la KSK (y el DS en la superior) – menos interacción administrativa

DNSSEC: RRSIG

- Resource Record Signature
  Lista las firmas hechas con la ZSK sobre un RRset



DNSSEC: RRSIG

- Valores por defecto típicos:
  La hora de creación de la firma es *1 hora antes*
  La caducidad de la firma es *en 30 días*
  Por supuesto, se recomienda seriamente el control apropiado de los temporizadores (NTP)
- Qué pasa cuando las firmas caducan?
  SERVFAIL...
  Su dominio desaparece de la Internet para los servidores que hacen validación
- Fíjese que *las claves no caducan*.
- Por lo tanto, el firmado *regular* es parte del proceso operativo (no sólo cuando hay cambios)
  La zona completa no se tiene que firmar...

DNSSEC: NSEC/NSEC3

- NSEC – comprobación de no-existencia
- Recuerde, los servidores autorizados están sirviendo records pre-calculados.

  NSEC provee un apuntador al **N**ext **SEC**ure record (próximo récord firmado) en la cadena de récords.

  ➔"no hay récords entre estos dos récords", firmado.

  Toda la zona está ordenada lexicográficamente:

```
myzone.
sub.myzone.
test.myzone.
```

DNSSEC: NSEC/NSEC3

```
myzone. 10800 NSEC test.myzone. NS SOA RRSIG NSEC DNSKEY

myzone. 10800 RRSIG NSEC 5 1 10800 20090317182441 (
                    20090215182441 5538 myzone.

                    ZTYDLeUDMlpsp+IWV8gcUVRkIr7KmkVS5TPH
                    KPsxgXCnjnd8qk+ddXlrQerUeho4RTq8CpKV
                     ...
                    )
```

- El último récord NSEC apunta de vuelta al primero.
- Problema:

  Enumeración de zonas (trazar la lista de récords NSEC)
  Sí, el DNS no debería usarse para publicar datos sensibles, pero las políticas varían.

DNSSEC: NSEC/NSEC3

- Si el servidor responde NXDOMAIN:

  Uno o más récords NSEC indican que el nombre (o el wildcard) no existe

- Si la respuesta es NOERROR:

  ...y la sección de respuesta está vacía

  ➔El NSEC comprueba que el TIPO no existe

DNSSEC: NSEC/NSEC3

- Y qué hay de NSEC3 ?

  No entraremos en detalles, pero el asunto es:

  - No firmar el nombre del record seguro siguiente, pero un ***hash*** de éste
    - Todavía es posible probar la no-existencia, *sin* revelar el nombre.
  - Esto es una explicación simplificada. El RFC 5155 explica los NSEC3 en *53* páginas.

  También introduce el concepto de “no participar” (vea la sección 6 del RFC) el cual tiene aplicación en las llamadas zonas “delegation-centric” con delegaciones sin firmar – en resumen: No pierda tiempo firmando RRsets para delegaciones que se conoce que no implementan DNSSEC.

DNSSEC: DS

- Delegation Signer
- Hash del KSK de la sub-zona
- Almacenado en la zona superior, junto con el record NS que indica la delegación de la sub-zona
- Los records DS de la sub-zona se firman *junto con el resto* de los datos de la zona superior

  Los records NS *NO* se firman (son una referencia)

```
myzone. DS 61138 5 1
 F6CD025B3F5D030408950535 4A0115584B56D683
myzone. DS 61138 5 2
 CCBC0B557510E4256E88C01B0B1336AC4ED6FE08C826
 8CC1AA5FBF00 5DCE3210
```

Digest type 1 = SHA-1, 2 = SHA-256

```
digest = hash( canonical FQDN on KEY RR | KEY_RR_rdata)
```

DNSSEC: DS

- Dos hashes generados por defecto:

1	SHA-1	MANDATORY
2	SHA-256	MANDATORY

- Hay nuevos algoritmos que están en proceso de estandarización
- Esto ocurrirá continuamente a medida que los algoritmos se determinen inseguros

DNSSEC: nuevos campos

- Hay cambios a nivel del paquete
- Los DNS recursivos que no soporten DNSSEC *deberían* ignorar los siguientes:

  CD: Checking Disabled (indicar al servidor recursivo que no haga validación, incluso si las firmas DNSSEC están presentes y se pueden verificar)

  AD: Authenticated Data, se marca en la respuesta del servidor con validación si la respuesta es segura, y el cliente pidió validación

- Opción nueva: EDNS0
  DO: DNSSEC OK (encabezado EDNS0 OPT) para indicar al cliente el soporte de las opciones DNSSEC

Demostración: los nuevos récords

Estatus de la seguridad de los datos

(RFC4035 § 4.3)

- Seguro

  El servidor recursivo es capaz de crear una cadena de records DNSKEY y DS firmados desde un punto de referencia seguro hasta el RRset

- Inseguro

  El servidor recursivo sabe que no hay una cadena de records DNSKEY y DS firmados desde ningún punto de referencia seguro hasta el RRset

- Falso (Bogus)

  El servidor recursivo determina que debería ser capaz de establecer la cadena de confianza, pero no es posible.

  Puede indicar un ataque, pero también puede ser un error de configuración o el resultado de la corrupción de algún dato

- Indeterminado

  El servidor recursivo no puede determinar si el RRset debería estar firmado

Firmar una zona...

Habilitar DNSSEC

- **Involucra a múltiples sistemas**

  Resolvers finales (en el sistema operativo)
  - Nada que hacer... pero hablaremos de eso luego

  Servidores caching (recursivos)
  - Habilitar la validación DNSSEC
  - Configurar las claves del dominio raíz

  Servidores autorizados
  - Habilitar la lógica de DNSSEC (si hace falta)
    - Firmar y servir datos no tiene que ser necesariamente en la misma máquina
    - El sistema que firma puede estar *offline*

Firmar la zona

1. Generar los pares de claves
2. Incluir los récords DNSKEYs públicos en el contenido de la zona
3. Firmar la zona con la clave secreta ZSK
4. Publicar la zona
5. Enviar los récords DS a la zona superior
6. Esperar...

1. Generar las claves

```
# Generar la ZSK
dnssec-keygen -a rsasha1 -b 1024 -n ZONE  myzone

# Generar la KSK
dnssec-keygen -a rsasha1 -b 2048 -n ZONE -f KSK myzone
```

Esto generó 4 ficheros:

- Kmyzone.+005+*id_of_zsk*.key
- Kmyzone.+005+*id_of_zsk*.private
- Kmyzone.+005+*id_of_ksk*.key
- Kmyzone.+005+*id_of_ksk*.private

2. Incluir las claves en la zona

Incluya los records DNSKEY para la ZSK y KSK en la zona, para que sean firmados con el resto de los datos:

```
cat Kmyzone*key >>myzone
```

O utilice la instrucción $INCLUDE para que sean cargados al leer la zona:

```
$INCLUDE “Kmyzone.+005+id_of_zsk.key”
$INCLUDE “Kmyzone.+005+id_of_ksk.key”
```

3. Firmar la zona

Firme su zona

```
# dnssec-signzone myzone
```

- *dnssec-signzone* usará todos los valores por defecto para la duración de la firma, el número de serie no será incrementado, y las claves privadas para firmar serán determinadas automáticamente.
- Firmar hará lo siguiente:

  Ordenar los datos (lexicográficamente)

  Insertar:
  - Récords NSEC
  - Récords RRSIG (firma de cada RRset)
  - Récords DS de ficheros *key-set* de sub-zonas (para la zona superior)

  Generar ficheros key-set y DS-set, para enviar a la zona superior

3. Firmar la zona (2)

- Desde la versión 9.7.0, BIND puede firmar y re-firmar sus zonas automáticamente

  Facilita las cosas significativamente

  Pero la generación de las claves, la gestión y la renovación aún tienen que hacerse por separado

4. Publicar las zonas firmadas

- Para publicar la zona firmada es necesario configurar el servidor para que cargue el fichero de la zona firmada.
- ... pero usted aún tiene que enviar sus récords DS de manera segura a la zona superior, de lo contrario, nadie sabrá que usted está firmando su zona con DNSSEC

5. Enviar el récord DS a la zona superior

- Es necesario enviar, de manera segura, el récord DS derivado de la KSK a la zona superior
  RFCs 4310, 5011
- ... pero y si la zona superior no usa DNSSEC?

Habilitar DNSSEC en el servidor recursivo

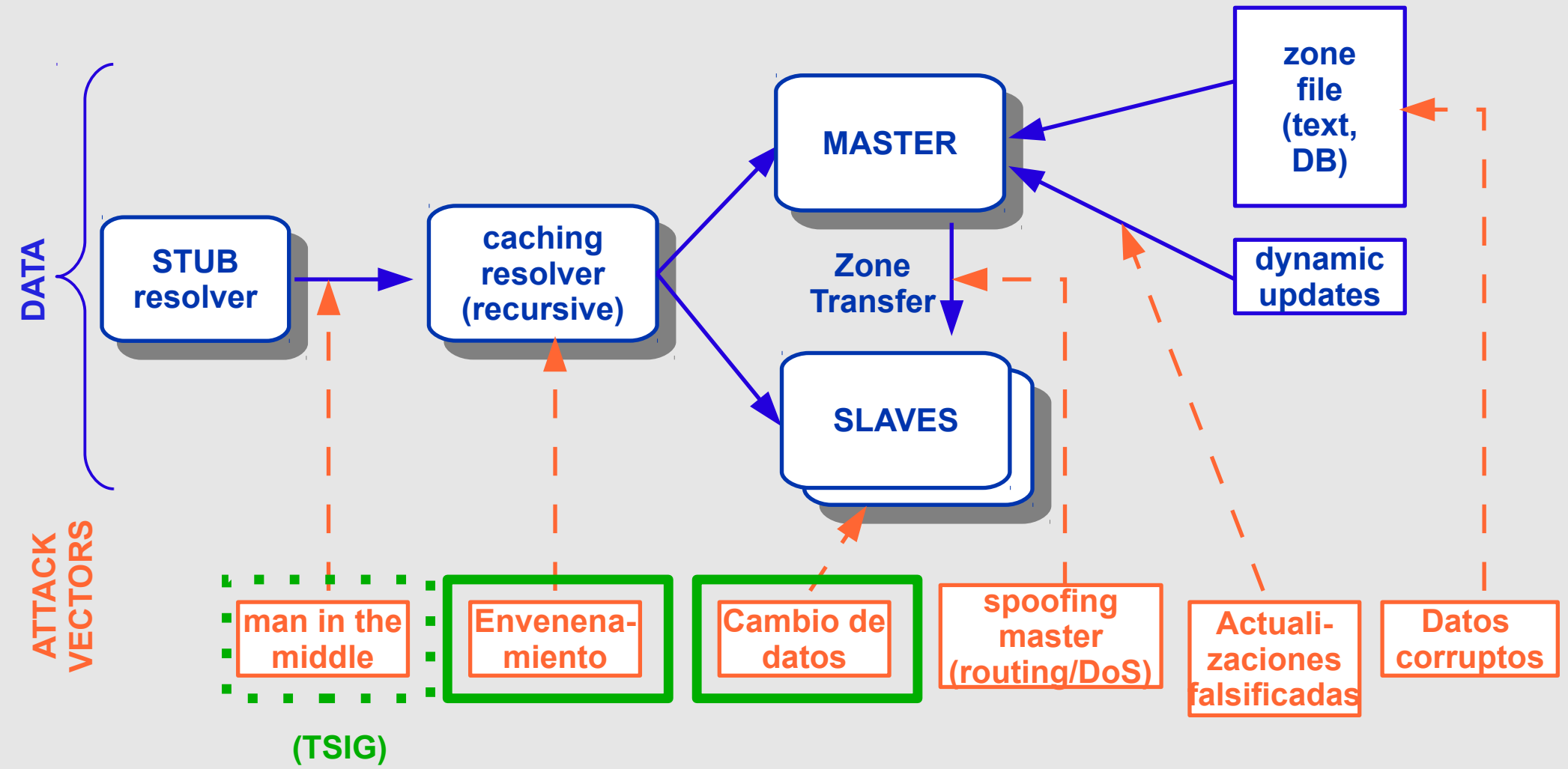
- Configure su servidor recursivo para validar DNSSEC
  No es estrictamente necesario, pero útil si quiere saber si su zona está siendo validada correctamente
- Probar...
- Recuerde, la validación se hace sólo en el servidor recursivo.

Resumen

- Generar las claves
- Firmar y publicar la zona
- Configuración del recursivo
- Probar la zona firmada

Alguna duda hasta ahora ?

Entonces, qué protege DNSSEC?



Qué NO protege?

- Confidencialidad
  Los datos no se cifran
- Comunicación entre el resolver (i.e: su sistema personal) y el servidor recursivo/caching.
  Para esto, tendría que usar TSIG, SIG(0), o tendría que confiar en su recursivo
  Éste último realiza la validación por usted

Por qué se ha tardado tanto en implementar ?

Muchas razones...

Es ”complicado”. Poca experiencia en la comunidad.
Hay más y más herramientas. La experiencia operacional es clave.
Riesgos de fallo (fallo en la firma, fallo en la renovación) lo cual hace que la zona desaparezca
La especificación ha cambiado varias veces desde los años 90
NSEC permite el listado de la zona.
Antes de Kaminsky, DNSSEC era considerado como *una solución en busca de un problema*
Tardanza en firmar la zona raíz (política)

Delegación de la autorización de firmar

Siguiendo la cadena de confianza (cortesía de RIPE)

Locally Configured

```
Trusted Key . 8907
```

(root) .

```
.                   DNSKEY (…) 5TQ3s… (8907) ; KSK
                    DNSKEY (…) lasE5… (2983) ; ZSK

                    RRSIG  DNSKEY (…)  8907 .  69Hw9…

org.                DS   7834 3 1ab15…
                    RRSIG   DS (…) . 2983
```

org.

```
org.                DNSKEY (…) q3dEw… (7834) ; KSK
                    DNSKEY (…) 5TQ3s… (5612) ; ZSK

                    RRSIG  DNSKEY (…)  7834 org.  cMas…

nsrc.org.           DS   4252 3 1ab15…
                    RRSIG  DS (…) org. 5612
```

nsrc.org.

```
nsrc.org.           DNSKEY (…) rwx002…  (4252) ; KSK
                    DNSKEY (…) sovP42…  (1111) ; ZSK

                    RRSIG  DNSKEY (…) 4252 nsrc.org.  5t...

www.nsrc.org.       A 202.12.29.5
                    RRSIG  A  (…)  1111 nsrc.org.  a3...
```

Despliegue de DNSSEC y Operaciones

Caducidad de las firmas

- Las firmas caducan en 30 días por defecto (BIND)
- Es necesario firmar regularmente:
  Para mantener una ventana constante de validez para las firmas de un RRset *existente*
- Para firmar *RRsets nuevos* y *actualizados*

- Quién hace esto ?

- Las claves en sí NO caducan...
  Pero sí necesitan ser renovadas...

Renovación de las llaves

- Tratar de minimizar el impacto
  - Validez corta para las firmas
  - Renovación regular de llaves
- Recuerde: Las DNSKEYs no tienen tiempo de vida
  - Los RRSIG de las DNSKEY sí tienen un tiempo de vida
- La renovación de claves requiere involucrar a otras entidades:
  - Se debe mantener el estado durante la renovación
  - Operacionalmente costoso
- Hay un estándar para esto: RFC5011 – BIND 9.7 lo soporta
- Vea http://www.potaroo.net/ispcol/2010-02/rollover.html

Renovación de claves

- Dos métodos para renovar las claves

  Pre-publicación
  Doble firma

- Se usan diferentes métodos para la renovación de KSK y ZSK
  (cortesía DNSSEC-Tools.org)

Renovación de claves

- **Renovación de ZSK usando el método de pre-publicación**

1. Publicar una ZSK nueva (junto con la actual)
2. Esperar a que las cachés obtengan el DNSKEY RRSet conteniendo la nueva ZSK (TTL)
3. Firmar la zona con la ZSK publicada
4. Esperar a que los datos obsoletos de la zona caduquen en las cachés (TTL)
5. Eliminar la ZSK anterior

Renovación de claves

- **Renovación de la KSK usando el método de doble firma**

1. Generar una nueva KSK
2. Firmar las DNSKEYs con ambas KSKs
3. Agregar los nuevos records DS a la zona superior
4. Esperar a que las cachés actualicen las DNSKEYs
5. Firmar las DNSKEYS con la KSK nueva solamente
6. Esperar a que las viejas DNSKEYs caduquen
7. Eliminar los DS obsoletos en la zona superior
8. Eliminar la KSK obsoleta de la zona

Herramientas automáticas

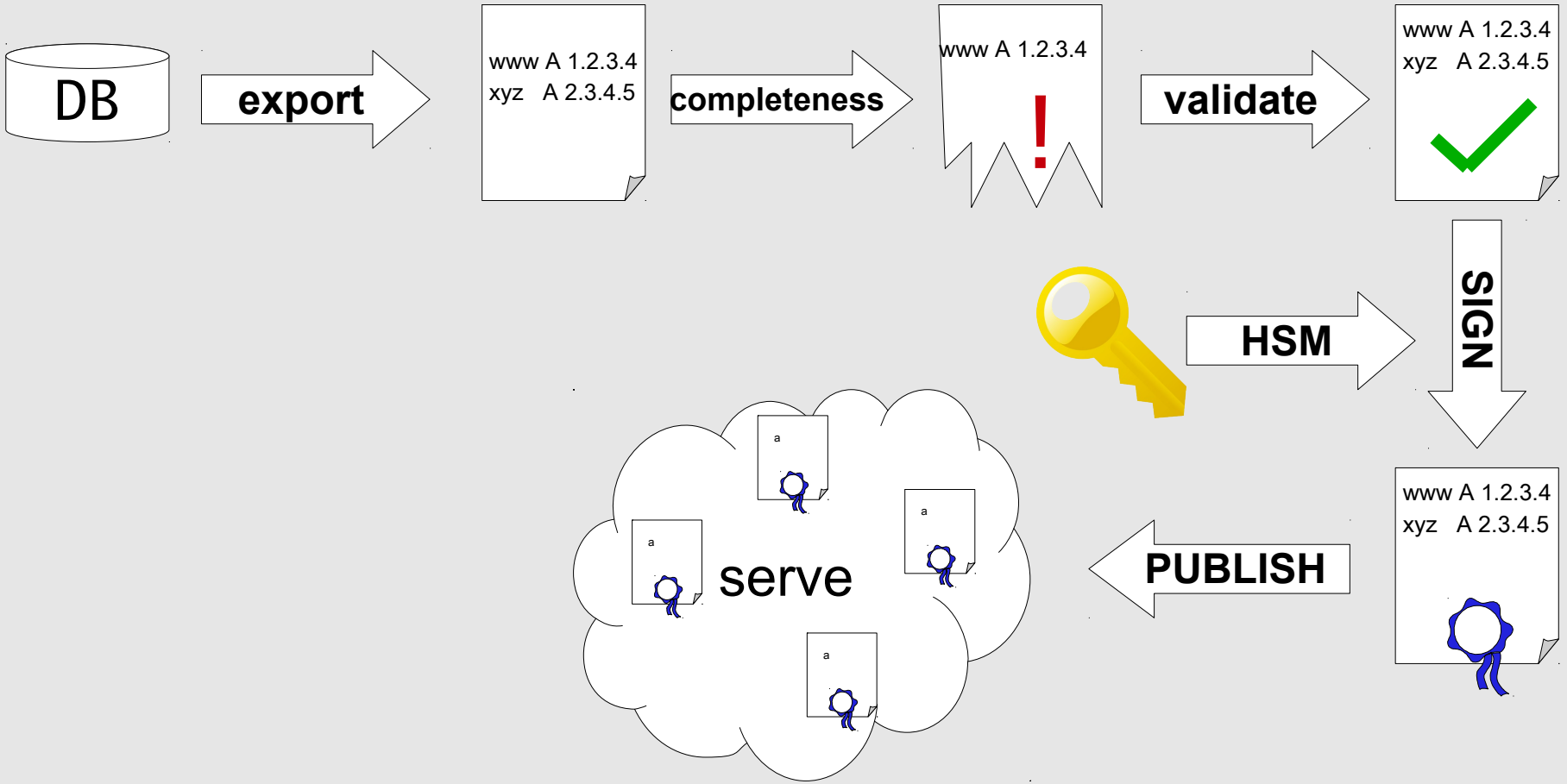
- Por suerte, ya existe un grupo de herramientas para hacer las operaciones de DNSSEC más fáciles
- No resuelven todos los problemas aún, como la interacción entre zona superior e inferior (gestión de DS), pero se encargan de todas las complicaciones de mantener una PKI (sí, esto es lo que es...)
- http://www.dnssec.net/software
  - www.opendnssec.org
  - www.dnssec-tools.org
  - http://www.ripe.net/projects/disi/dnssec_maint_tool/
  - http://www.hznet.de/dns/zkt/
  - …

Qué es necesario para desplegar DNSSEC ?

- Un DPS (Declaración de políticas y prácticas de DNSSEC)
  http://tools.ietf.org/html/draft-ietf-dnsop-dnssec-dps-framework-03
  Detalla el diseño, implementación, métodos y prácticas que dictan la operación de una zona firmada con DNSSEC
  Ayuda a que entidades externas revisen el proceso y evalúen la confiabilidad del sistema.
- Una comprensión profunda de DNS
- Un marco de referencia operacional en el cual insertar el proceso de DNSSEC
  Mucha más probabilidad de fallar si la organización no tiene buenos procedimientos en un principio

Qué es necesario para desplegar DNSSEC ? (2)

- Monitorización



Seguridad física

- HSM – Hardware Security Module
  Módulo de seguridad basado en hadware

Explicación breve

Problemas con el despliegue y otras cosas

Falta de experiencia operacional...

Todo el mundo habla sobre DNSSEC

- ... pero poca gente tiene experiencia práctica en la operación del día a día
- DNSSEC no se puede simplemente encender y apagar

  Dejar de firmar una zona no es suficiente
  La zona superior debe dejar de publicar los récords DS y las firmas
- Los tipos de fallos son bien conocidos,pero los procedimientos de recuperación son difíciles

Mecanismos de publicación de Récords DS

Comunicación del DS a la zona superior estandarizada, pero no común, o se usan otros métodos

- Subir via web con SSL ?
- E-mail con PGP/GPG ?
- Extensiones EPP (RFC4310)

- Recuerde, esta comunicación debería ser confiable
- Re-delegación o cambio del registrador cuando la zona está firmada
  - Compartir la clave durante la transición ?
  - Apagar DNSSEC de momento ?
  - Qué pasa si el administrador original no coopera ?
  - ➔ Problemas de políticas

EDNS0 y firewalls o servidores DNS defectuosos

DNSSEC requiere EDNS0

Paquetes con más información > 512 bytes
Los firewalls no siempre reconocen EDNS0
TCP filtrado – administradores sobre-protectores

- Muchas redes de hoteles (quizá incluso éste mismo) no permiten pasar récords DNSSEC

Aplicaciones que no entienden de DNSSEC

Esto podría ir para largo...

- No existen prácticamente aplicaciones que hagan uso de DNSSEC

  El usuario no puede saber qué está fallando
  Las preguntas le caen al administrador de la red
  - ➔ Compare con fallos de SSL (para usuarios que pueden leer...)
- Existen APIs – hasta ahora 2
  - http://tools.ietf.org/id/draft-hayatnagarkar-dnsext-validator-api-07.txt
  - http://www.unbound.net/documentation/index.html

  - ➔ Firefox plugin (puente entre el nivel DNS y el usuario)
  - ➔ Qué pasa si las aplicaciones directamente marcan el bit +CD ?

Seguridad del último enlace

- Los *resolvers* finales aún son vulnerables a ataques tipo *man-in-the-middle*
  - No hay mucho qué hacer con eso
  - Confíe en su recursivo, o use TSIG
  - Cómo distribuir las claves ? (MS usa GSS-TSIG con Kerberos)
  - Los recursivos no están diseñados para lidiar con cientos de miles de clientes con TSIG
  - SIG(0) tampoco es eficiente a gran escala
- Se está trabajando en estos problemas
  - DNS sobre protocolos de transporte para evitar los filtrados excesivos
  - El proyecto dnssec-trigger