Sicurezza delle reti[1]

Mattia Monga

Dip. di Informatica e Comunicazione
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

a.a. 2010/11

Lezione XVIII: La protezione del DNS

La protezione del DNS

Il DNS è un servizio fondamentale per il buon funzionamento delle reti.

- È un elemento molto importante nella costruzione della catena di trust della maggior parte delle transazioni iniziate da un utente umano, che raramente usa direttamente i numeri IP
- È un servizio generalmente pubblico e ottenuto in maniera decentralizzata, quindi nessuno ne ha il completo controllo
- Può essere utilizzato anche come strumento di "intelligence" prima di ulteriori attacchi
- Esistono moltissime implementazioni, non tutte curate dal punto di vista della sicurezza
- Per questi motivi è un bersaglio particolarmente attraente

Separazione di servizi DNS

Spesso si allestiscono due server DNS

DNS Esterno Riceve query da utenti esterni per informazioni riguardo host pubblicamente accessibili della rete aziendale, incluso l'MX server (il Mail Relay)

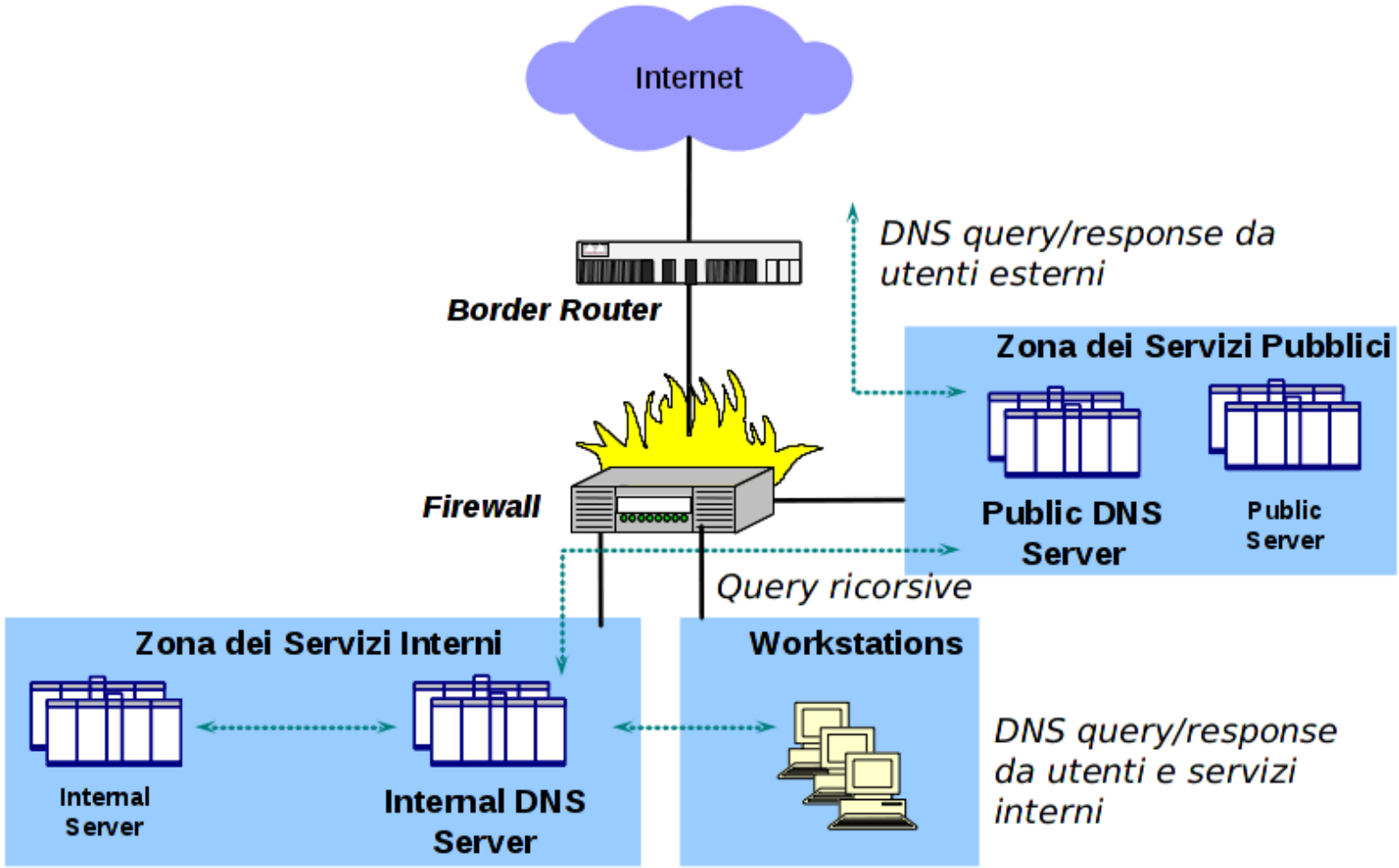
DNS Interno Riceve query da utenti interni per informazioni su host sia della intranet aziendale che di Internet. Per le query che il DNS Interno non è in grado di risolvere contatta altri DNS (query ricorsive).

Vantaggi della separazione

I due DNS mantengono informazioni differenti (solo quelle pubbliche l'esterno, tutte quelle della intranet l'interno) e hanno connessioni con zone a diverso grado di sicurezza.

- Separazione fisica delle informazioni riguardante servizi pubblici da quelle riguardanti servizi della intranet
- Assegnazione a diverse zone di sicurezza per la protezione delle informazioni
- Isolamento del DNS pubblico dalla rete interna nel caso di compromissione

Separazione DNS

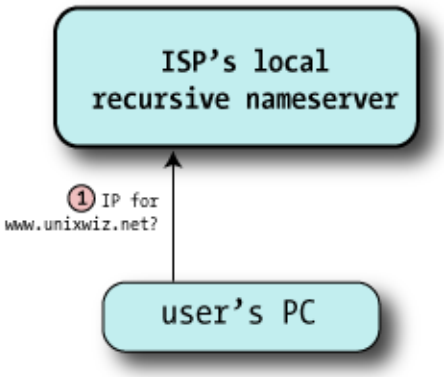


Risoluzione di una query

1. L'utente deve risolvere il nome `www.example.com` (per una connessione TCP o UDP serve il numero IP corrispondente)
2. Parte un richiesta al server DNS configurato nel sistema (p.es. via `/etc/host`) del record di tipo `A` associato a `www.example.com`
3. Il DNS locale esamina se si tratta di un indirizzo risolvibile localmente o se la query è *ricorsiva:* in questo caso consulta l'elenco dei (13) root server che conosce
4. Il root DNS non conosce l'indirizzo, ma risponde con un record di tipo `NS` corrispondente ai Global Top Level Domain (GLTD) server del dominio `.com`
5. La procedura si ripete coi GLTD, che rimanderanno al DNS autorevole (authoritative) per *example.com*

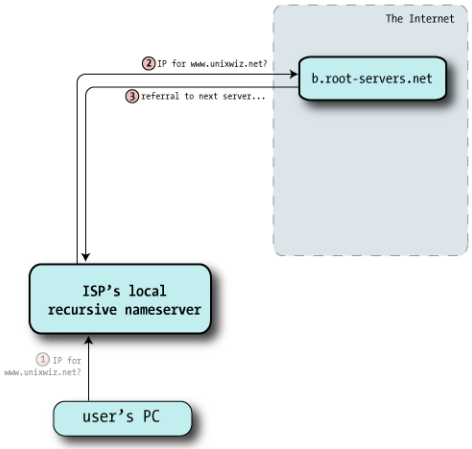
Risoluzione

da http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html



L'utente chiede la risoluzione di `www.unixwiz.net` al DNS del proprio ISP (`dnsr1.sbcglobal.net`)

Risoluzione



da http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

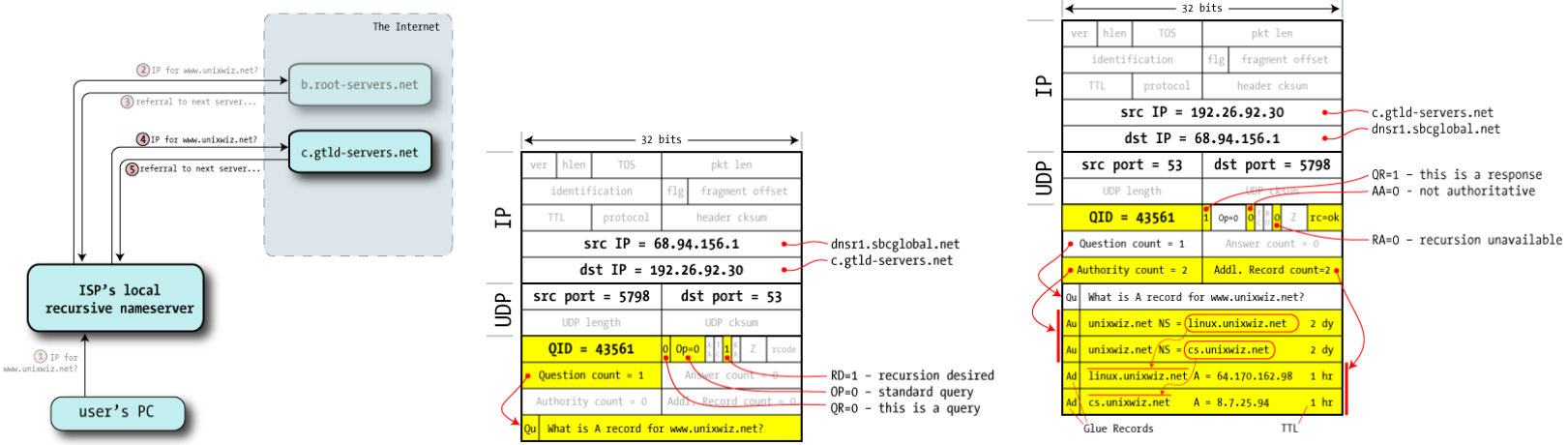
13 root server

```
A.ROOT-SERVERS.NET.   IN  A  198.41.0.4
B.ROOT-SERVERS.NET.   IN  A  192.228.79.201
C.ROOT-SERVERS.NET.   IN  A  192.33.4.12
...
M.ROOT-SERVERS.NET.   IN  A  202.12.27.33
```

e i name server di .net

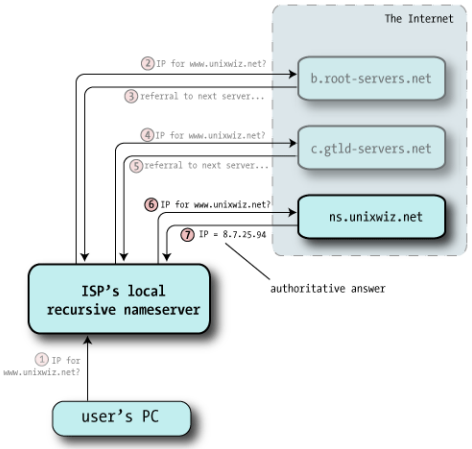
```
/* Authority section */
NET.                  IN  NS A.GTLD-SERVERS.NET.
                      IN  NS B.GTLD-SERVERS.NET.
                      IN  NS C.GTLD-SERVERS.NET.
                      ...
                      IN  NS M.GTLD-SERVERS.NET.

/* Additional section - "glue" records */
A.GTLD-SERVERS.net.   IN  A  192.5.6.30
B.GTLD-SERVERS.net.   IN  A  192.33.14.30
C.GTLD-SERVERS.net.   IN  A  192.26.92.30
...
M.GTLD-SERVERS.net.   IN  A  192.55.83.30
```

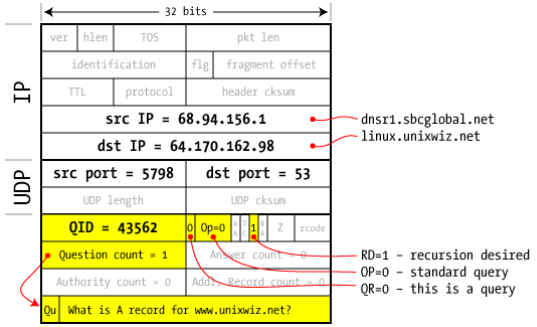
Risoluzione

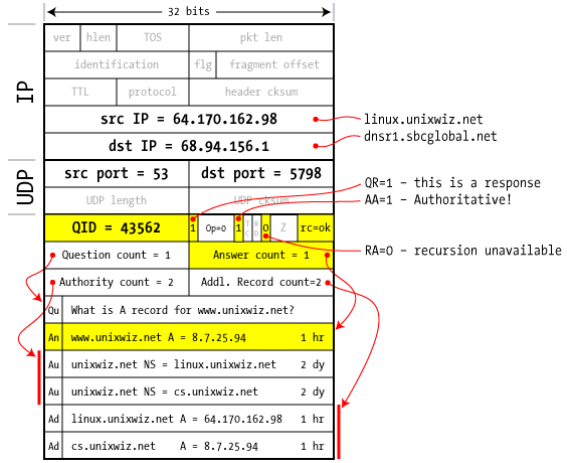


Risoluzione

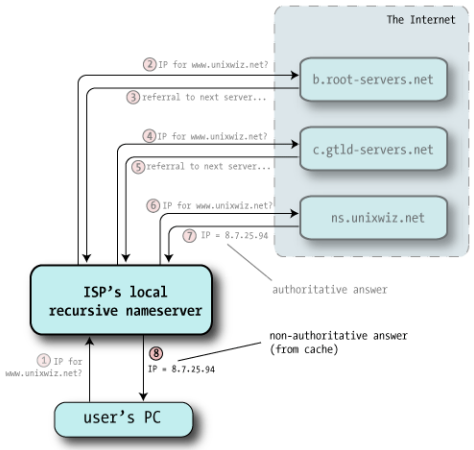
da http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html







Risoluzione



da http://www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

Il numero IP cercato è `8.7.25.94`. Questo dato può essere conservato (per un tempo pari al TTL) in una cache locale del name server ricorsivo per rendere piú efficiente il processo.

Fragilità del sistema

Il sistema è estremamente efficiente e piuttosto resistente ai guasti, ma nella versione originaria non prevede nessuna tecnica di autenticazione e integrità delle informazioni.

- Il name server *x* di `unixwiz.net` potrebbe ospitare le associazioni per i nomi della *zona* `bancaditalia.com`, anche se nessun GLTD ne delegherebbe la risoluzione a *x*
- Un attacco possibile è l'avvelenamento della cache (cache poisoning)

Cache poisoning

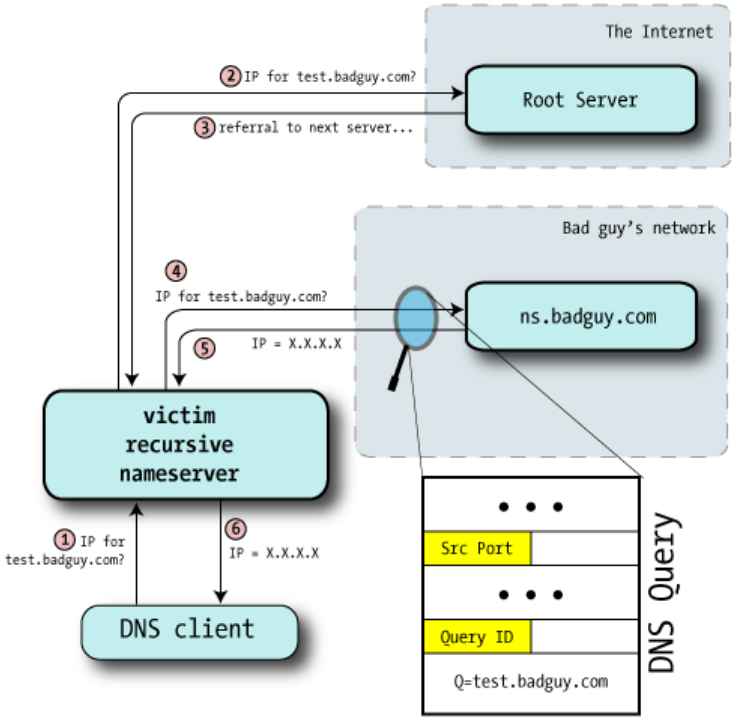
Un attaccante riesce ad alterare la cache di un DNS ricorsivo, che pertanto restituisce un'associazione scorretta.
Come fa il DNS ricorsivo ad "autenticare" la risposta che riceve da `ns.unixwiz.net`?

1. La risposta deve arrivare con la stessa porta UDP sorgente della richiesta. altrimenti viene scartata
2. La sezione Question coincide con quella della richiesta
3. Il query ID corrisponde a quello della richiesta
4. La risposta contiene dati riguardanti nodi nella zona (non `bancaditalia.com` per esempio)

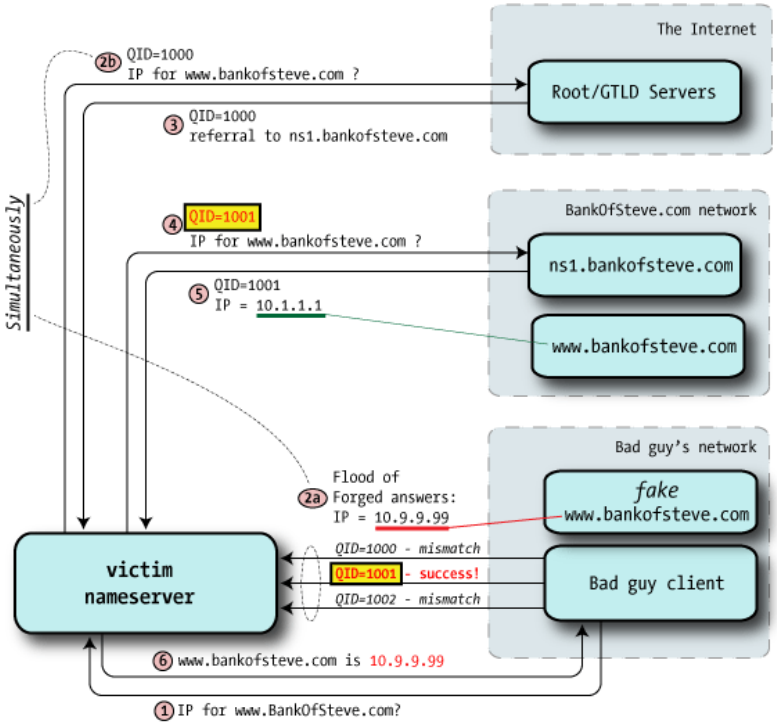
Se l'attaccante riesce a prevedere questi dati, è in grado di alterare la cache del DNS ricorsivo!

Come indovinare il Query ID?

Spesso è semplicemente un contatore, quindi basta intercettare il traffico di richieste lecite



Caso semplice



Se la porta UDP utilizzata è sempre la stessa (cosí in molti dns con scarsa attenzione alla sicurezza) l'attacco è semplice

Ovviamente non funziona se il nome è già nella cache. Le chance dell'attaccante sono minori se il dns authoritative è piú *vicino* al dns vittima.

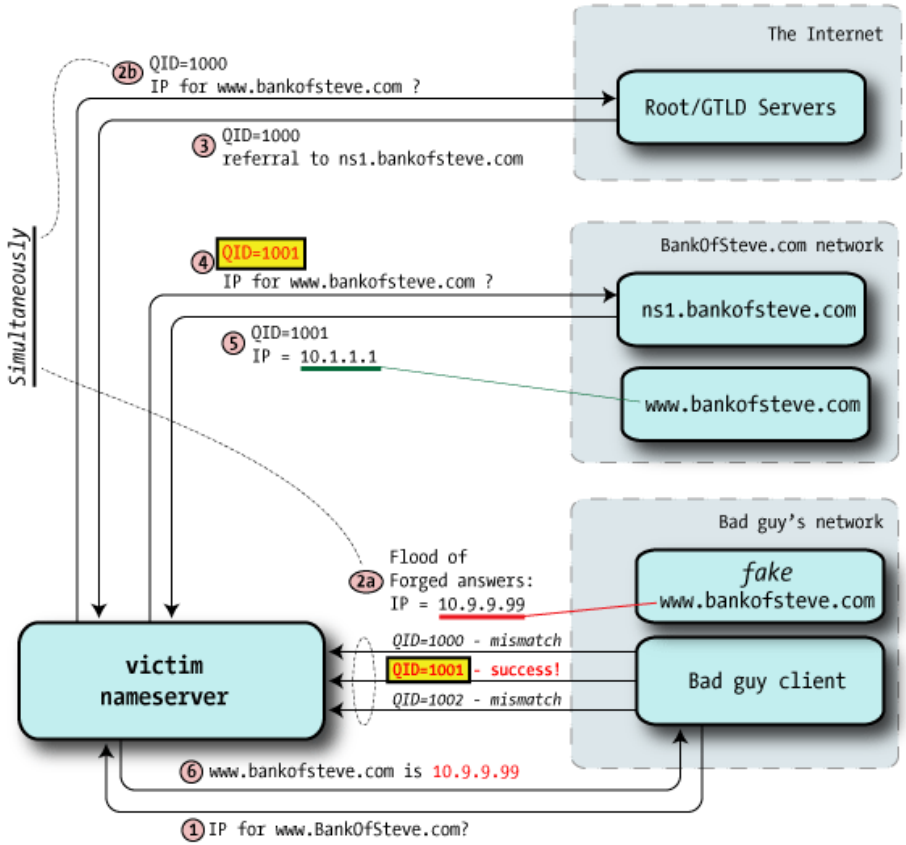
Difesa

La difesa principale è la randomizzazione del query ID

- Con ID sequenziali l'attaccante prova una ventina di ID
- Con ID random (su 16 bit) occorre provare 64K (prima che il dns authoritative risponda)

L'attacco di Dan Kaminsky

L'idea di base è la stessa, ma amplia l'impatto falsificando il dns authoritative stesso. L'attaccante ne allestisce uno proprio, che però normalmente non riceverebbe richieste.



Chance dell'attacco

Con la randomizzazione dei query ID sembra difficile fare le 64K prove necessarie in tempo utile (prima che arrivi la vera risposta).
Ma nella versione Kaminsky l'attaccante genera tanti nomi casuali (p.es. `www12345678.bankofsteve.com`). Anche se riesce a fare solo poche (50?) risposte finte prima di essere superato dal vero authoritative, può comunque ripetere questo tentativo tante volte con nomi diversi: ogni tentativo ha probabilità di riuscita $\frac{50}{65536}$
Con 100 tentativi: 7,3%, con 1000: 53,4%, con 10000: 99,9%
Un possibile miglioramento si ha randomizzando anche la porta UDP. Se la porta è random su 65535 ci vogliono almeno $60 \cdot 10^6$ tentativi. (MS DNS server sceglie fra 2500 porte, quindi in realtà "bastano" $2{,}3 \cdot 10^6$)

DNSSEC

DNSSEC è uno standard retro-compatibile che aggiunge autenticazione e controllo d'integrità alle query DNS. Prima versione 1997, rivisto nel 2005 e nel 2008.

- È considerato un elemento fondamentale nelle strategie globali della cosiddetta *trusted* Internet
- In realtà la sua adozione langue:
  - Complessità delle configurazioni
  - Aumento del traffico
  - Perplessità di una parte della comunità sull'efficacia

Nel 2009 risultavano 274 domini firmati su circa 80'000'000 di `.com`

DNSSEC, punti fondamentali

Il concetto fondamentale è che le risposte dei DNS authoritative sono firmate digitalmente

- La chiave pubblica di una zona viene distribuita dalla zona gerarchicamente superiore (la chiave pubblica di `.net` è distribuita da un root server, ecc.)
- C'è la possibilità di avere risposte di non esistenza ("Authenticated denial of existence")

Il deployment è reso complicato soprattutto dall'esigenza di ruotare le firme che scadono (di solito ogni 30 giorni), per evitare *replay attack*.

Limiti di DNSSEC

Secondo D. J. Bernstein, autore di `djbdns` e di una proposta alternativa (DNSCurve), DNSSEC è mal progettato perché:

- L'assunzione di base è che non è pensabile usare la crittografia in ogni pacchetto, per ragioni di efficienza.
- Come conseguenza
  - Non c'è crittografia dei dati (solo integrità)
  - Le firme sono precalcolate (e quindi occorre ruotare le chiavi per limitare i replay)
    - Tutti i tool per la modifica dei dati DNS devono essere 'signature aware'
  - Non c'è protezione contro DoS

Ulteriori pericoli

Bernstein identifica anche vulnerabilità di DNSSEC

1. Ogni pacchetto di risposta DNS è firmato: se i dati sono alterati andrebbe scartato
   - So che i dati potrebbero essere falsi, ma non ho comunque i dati veri (denial of service)
   - Di fatto, al momento la maggior parte dei server non lo fa
2. Vengono firmate solo le associazioni di cui un ns è authoritative: i glue record rimangono falsificabili
3. Il protocollo permette l'amplificazione di DDos (una query di 78 byte si può trasformare in una risposta da 3113 byte)