UNIVERSITÄT DUISBURG ESSEN

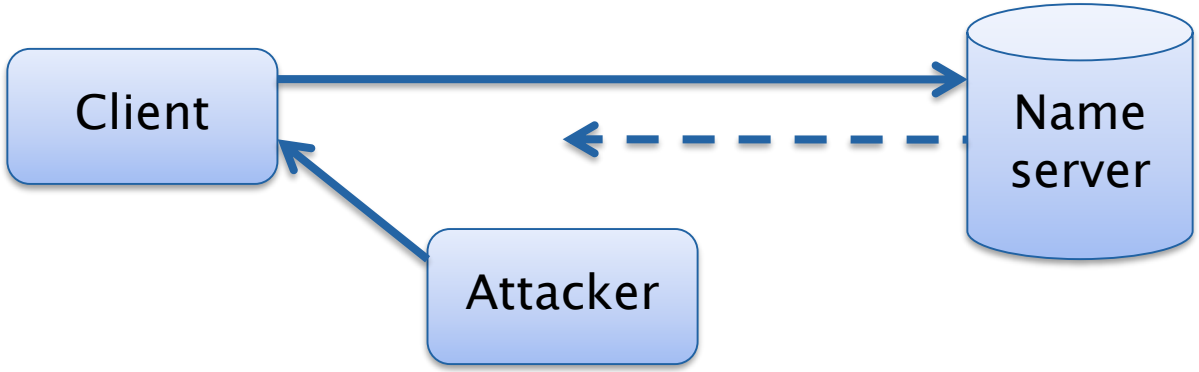
Internet Technology & Web Engineering

DNS Security

Dr.-Ing. Matthäus Wander

Universität Duisburg-Essen

Spoofing



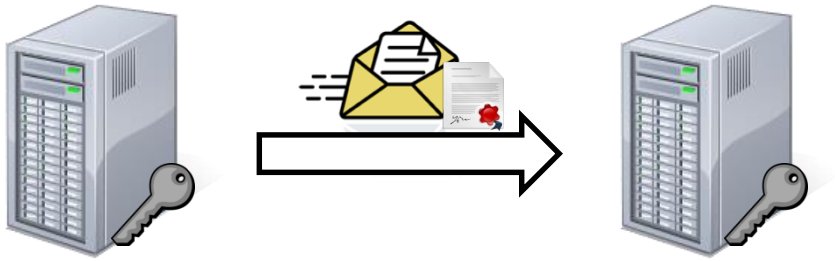
- Client sends DNS query over UDP
- Attacker spoofs response
  - Blocks nameserver response
  - Or simply responds faster
- How does the attacker know the query content?
  - On-path: by listening „on the wire“ (sniffing)
  - Off-path: by prediction or guessing

Wishlist: Desired Security Goals

- Data Integrity
  - Ensure messages are not altered during transmission
- Authenticity
  - Ensure the sender identity is trusted and known
- Confidentiality/Privacy
  - Ensure the message is accessible only to authorized receivers
  - Ensure the queried names are private
- Availability
  - Ensure DNS service is operable and working correctly

TSIG – Transaction Signature (1)

- Establishes Data Integrity and Authenticity
- But **not** confidentiality or availability
- "Secret Key Transaction Authentication for DNS" (TSIG)
  - Specified May 2000 (RFC 2845)
- Symmetric cryptography
- Shared key (sender and receiver use identical key)
- Secures messages with a Message Authentication Code
  - Point-to-point security
  - HMAC-SHA256



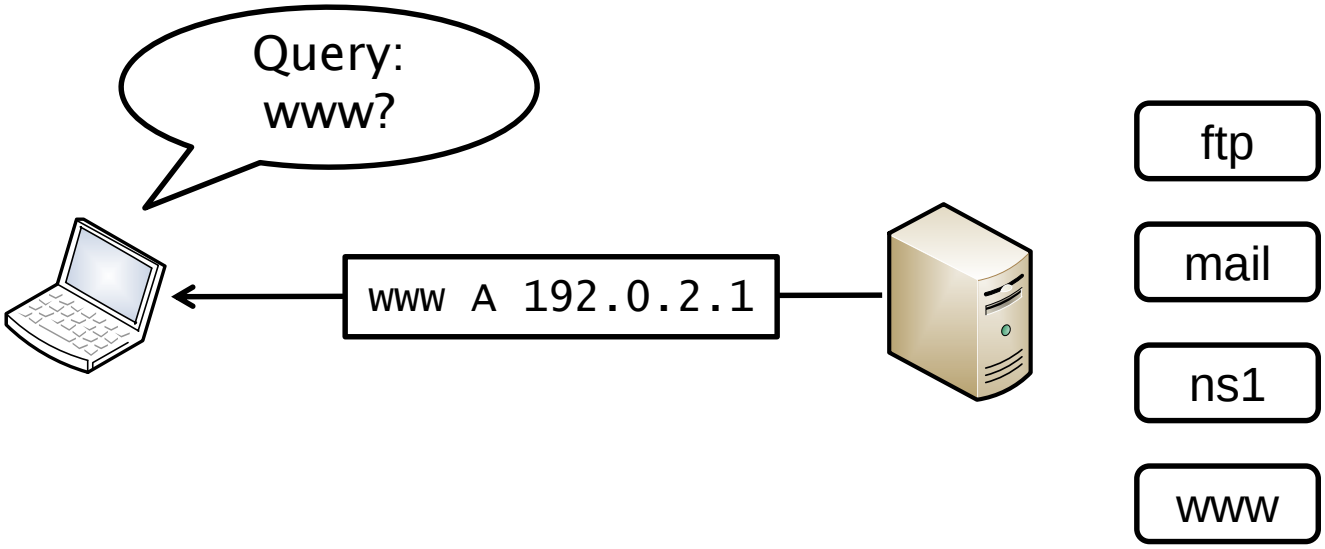
TSIG – Transaction Signature (2)

- Uses TSIG resource record for cryptographic parameters
  - „Meta“ record, not cached
  - Timestamp prevents replay attacks
- Key distribution manually over secure channel
  - Works only for small groups
  - e.g. zone transfer from primary to secondary nameserver
- TKEY enables secure key transfers (RFC 2930)
  - But requires an already secure DNS communication
  - Initial key still required to be transfered manually
- Not practical for global use

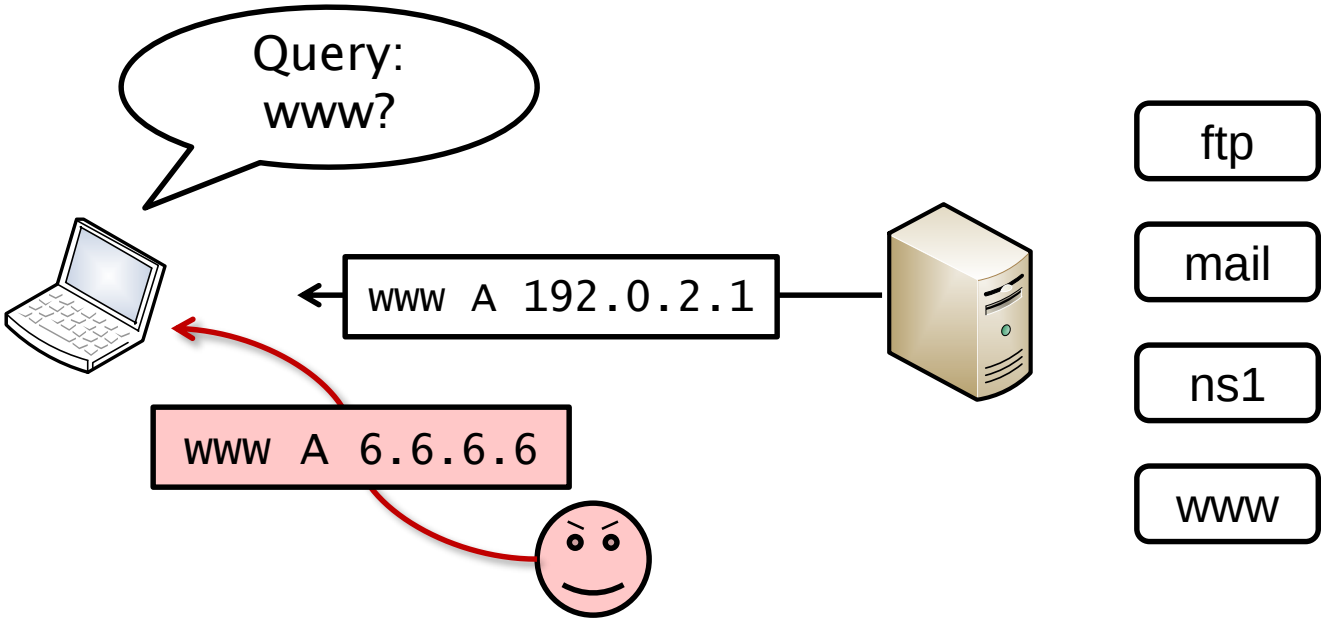
DNSSEC – Overview

- Establishes Data Integrity and Authenticity
- But **not** confidentiality or availability
- "Domain Name System Security Extensions" (DNSSEC)
  - Originally 1999 specified
  - Thrown away and built from scratch 2005 (RFCs 4033–4035)
- Asymmetric cryptography (RSA, DSA, ECDSA)
- Private and public key
- Requires EDNS for long DNS messages

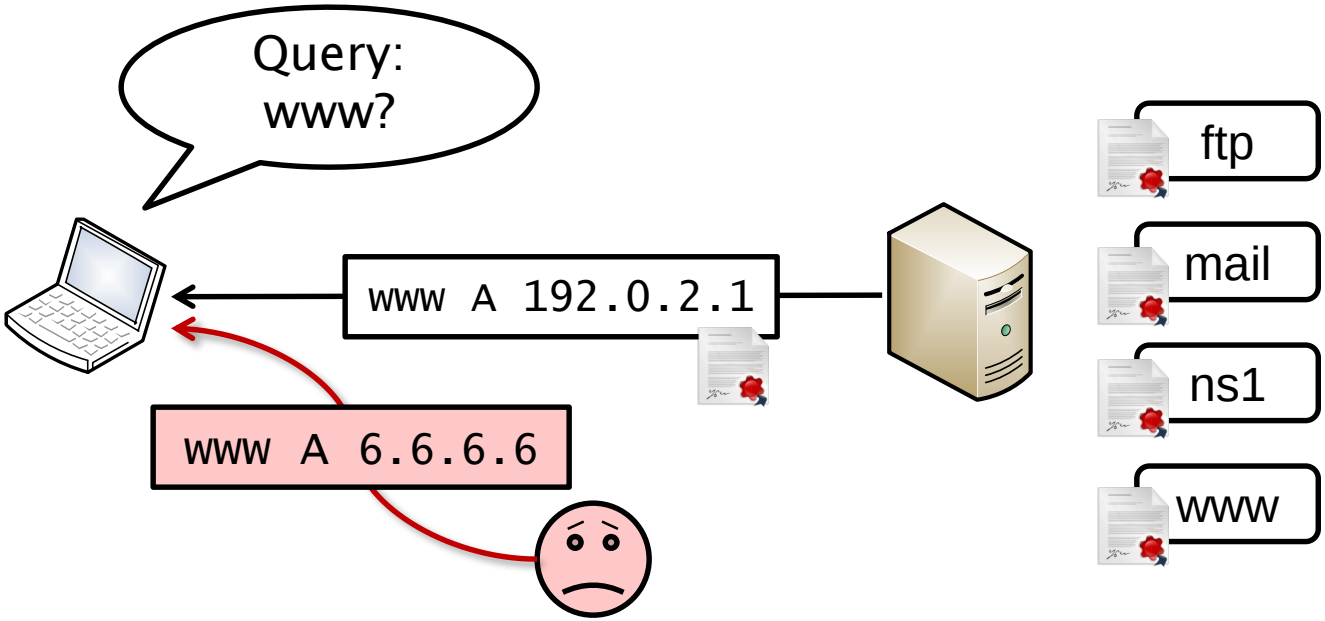
DNSSEC – Security Goal



DNSSEC – Security Goal



DNSSEC – Security Goal

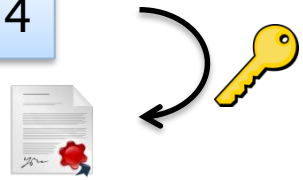


- Data integrity and authenticity
- Signatures over resource records (data sets)

DNSSEC – Concept

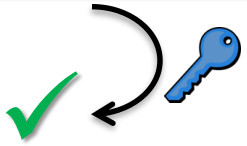
- Sign resource records with private key

```
example.net.    IN   A    1.2.3.4
```



- Signatures available as RRSIG records

```
example.net.    IN   RRSIG   A 5 3 600 20120519[…] eZMjxNZeX[…]
```



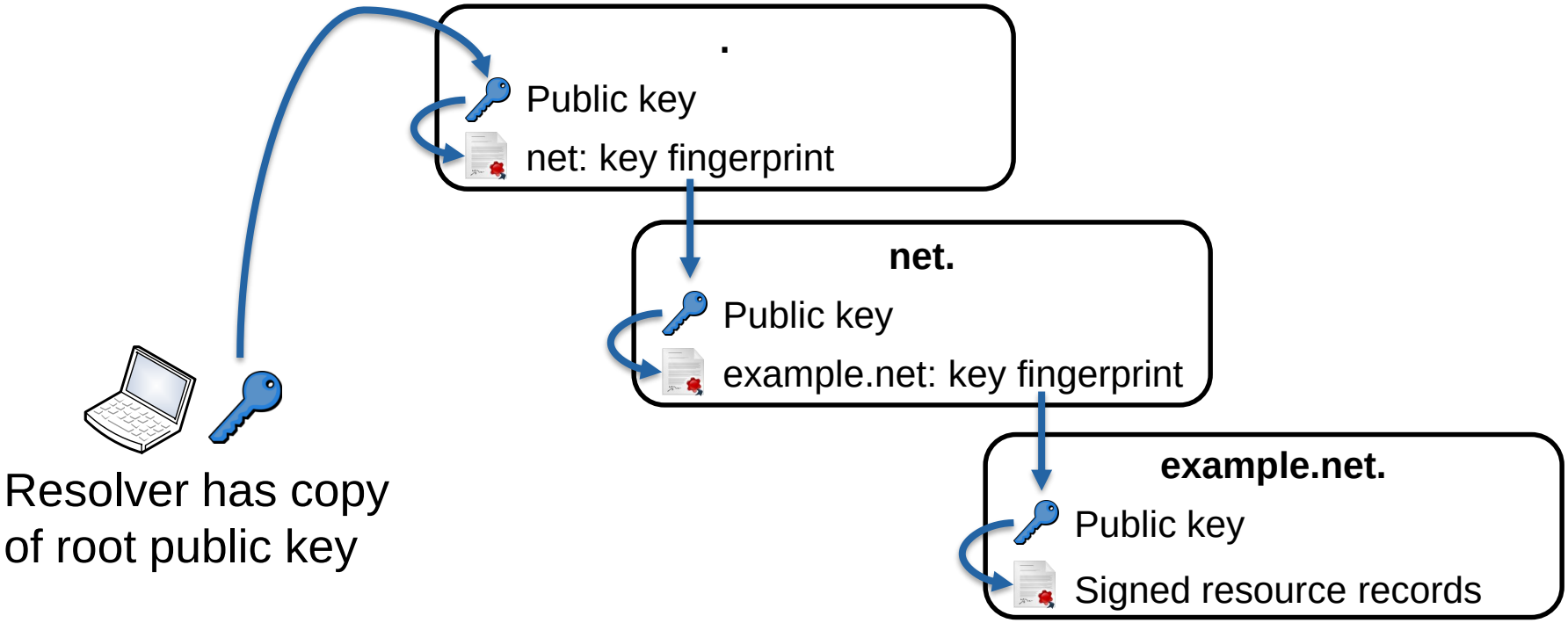
- Verify signature with public key
- Public key available as DNSKEY record

```
example.net.    IN  DNSKEY 256 3 8 AwEAAbd0IPTQdvyndWSX6H[…]
```

- How to verify authenticity of DNSKEY?
  - Tie DNSKEY with parent zone to create Authentication Chain

DNSSEC – Public Key Distribution

- Public keys distributed in-band
- Authenticated by parent domain
  - Fingerprint (hash value) of subdomain public key



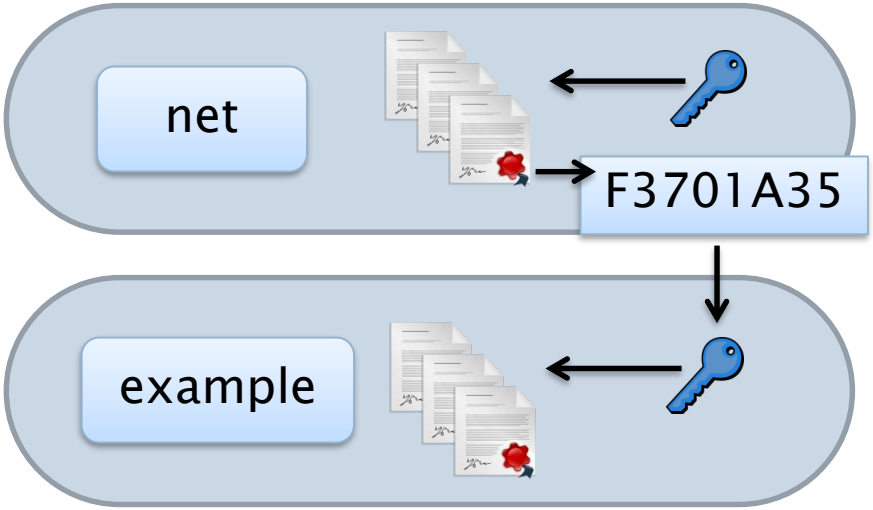
DNSSEC – Secure Delegation

- Compute hash (fingerprint) of DNSKEY

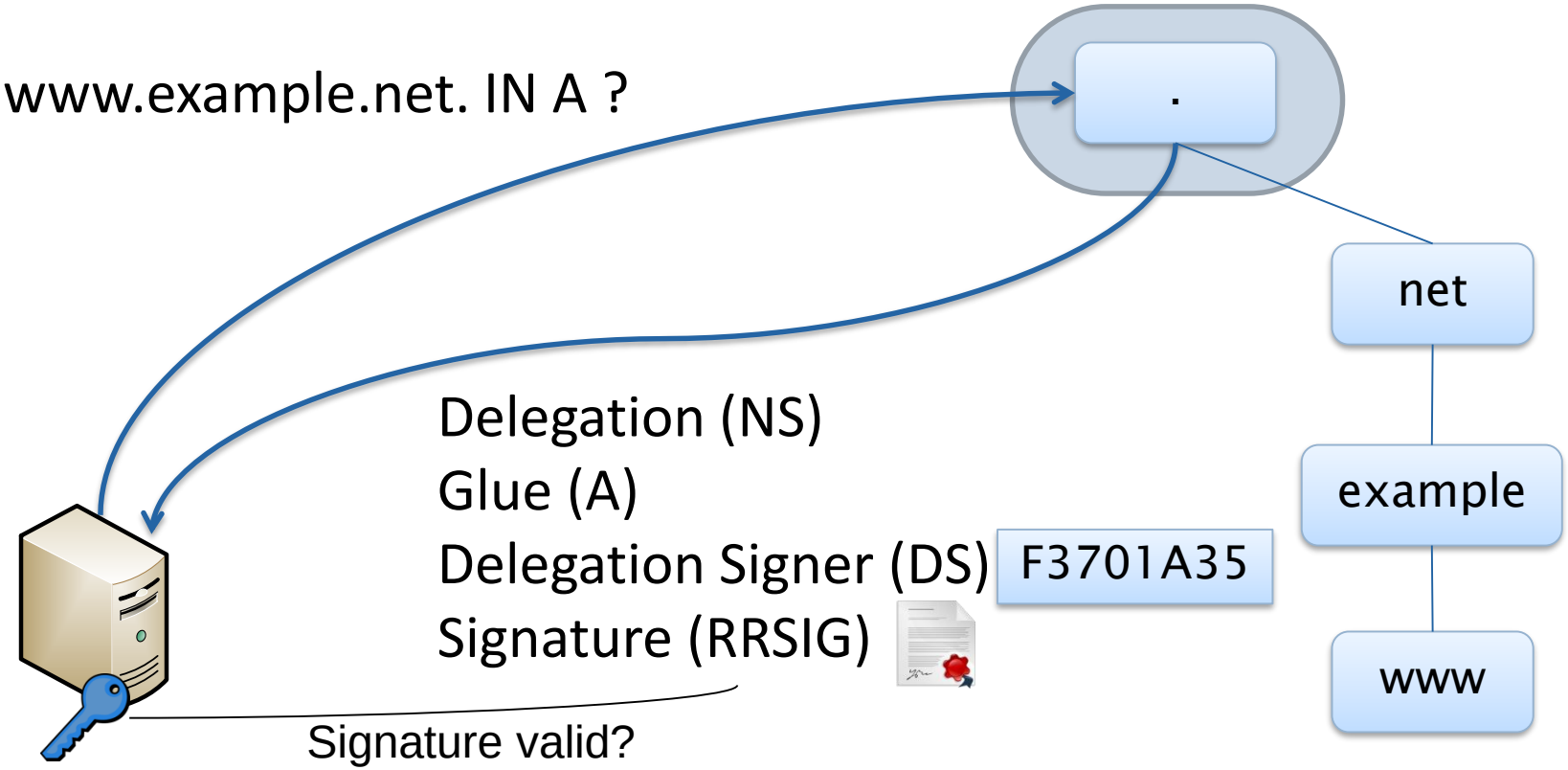
- Put hash as Delegation Signer (DS record) in upper zone

```
example.net.    IN   DS    12892 5 1 F3701A35[...]
```

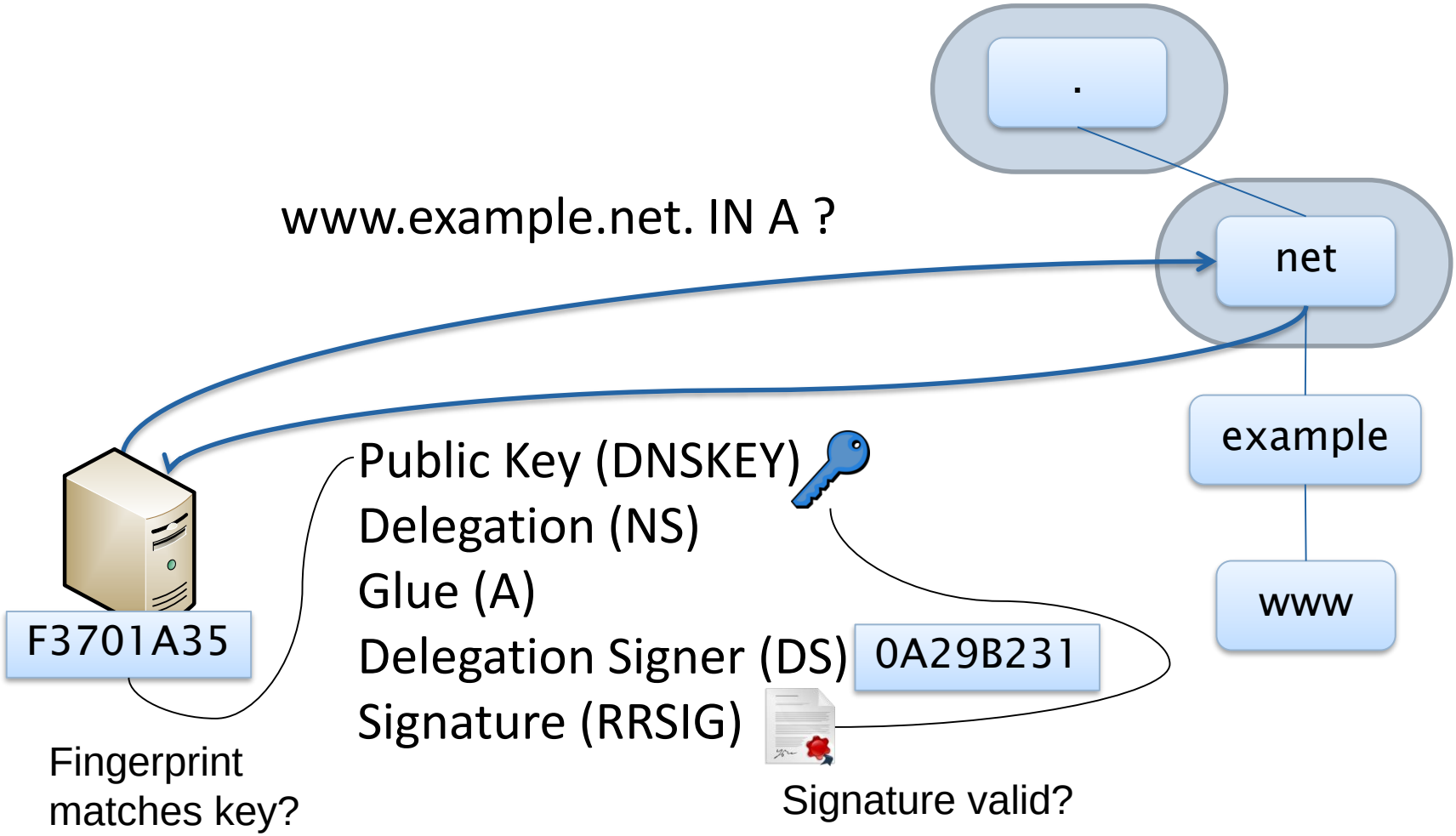
- Sign DS record with upper zone‘s DNSKEY



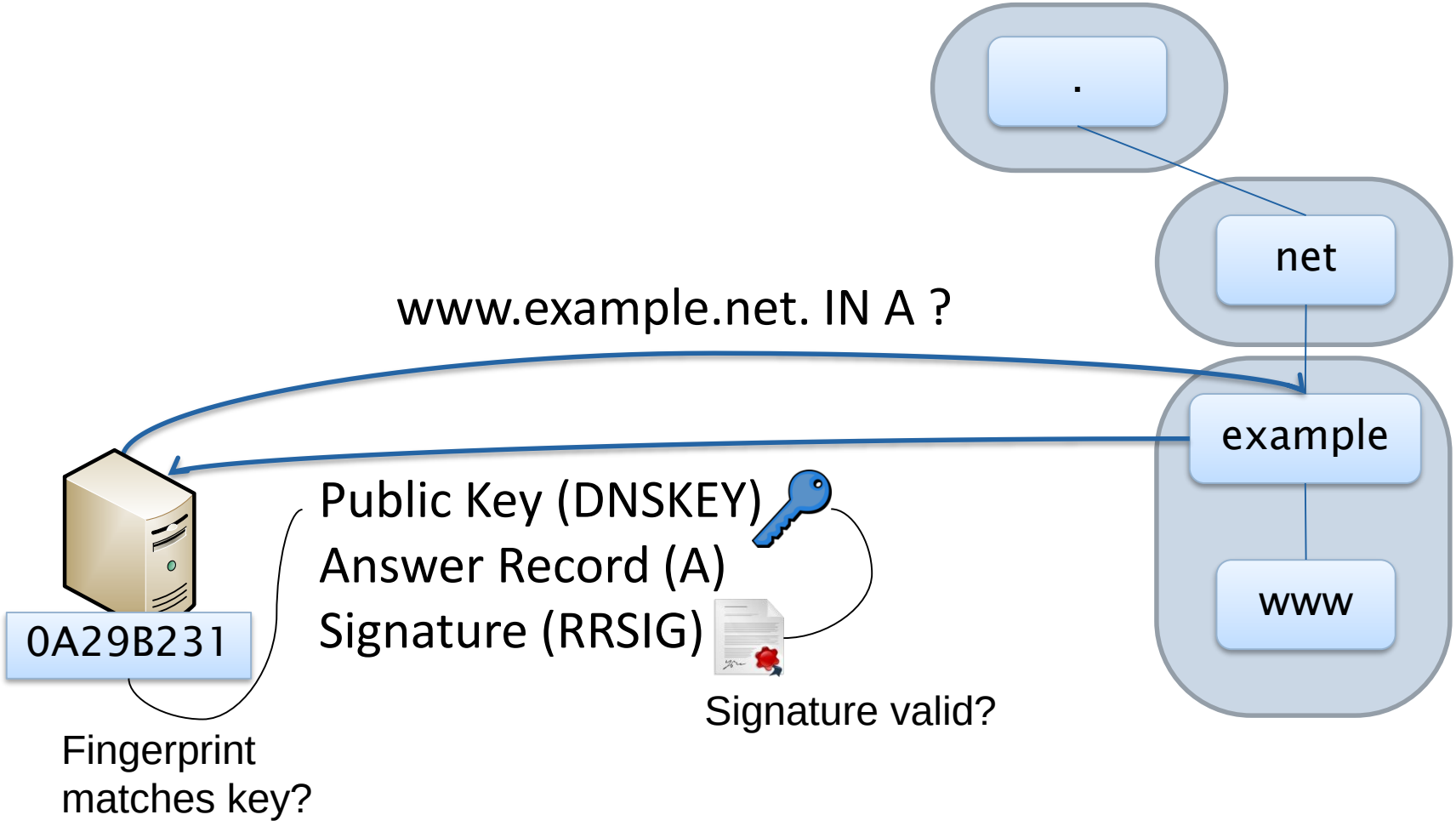
DNSSEC – Example Lookup (1)



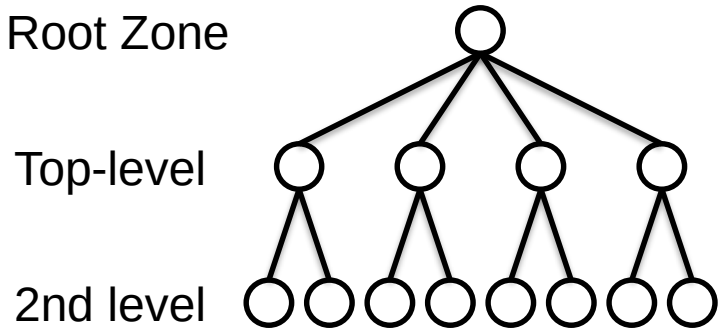
DNSSEC – Example Lookup (2)



DNSSEC – Example Lookup (3)

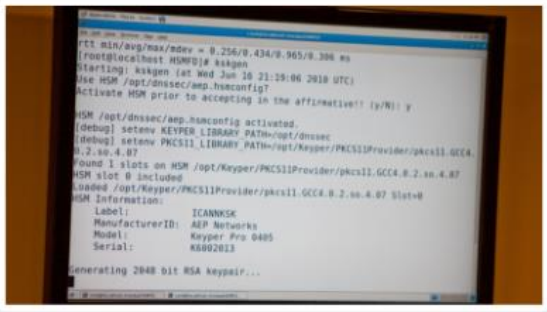


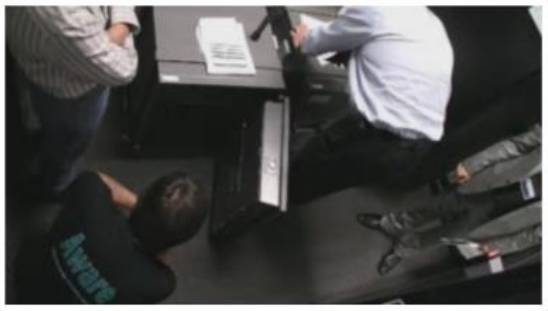
DNSSEC Trust Model



- Root zone uses two keys
  - Long-term Key Signing Key (KSK): signs the ZSK
  - 3-month Zone Signing Key (ZSK): signs zone data
- Root KSK shipped with DNSSEC validator software
  - Managed by ICANN (US-based non-profit organization)
  - 2048-bit RSA key, created in 2010
  - Planned to be replaced on October 11, 2018

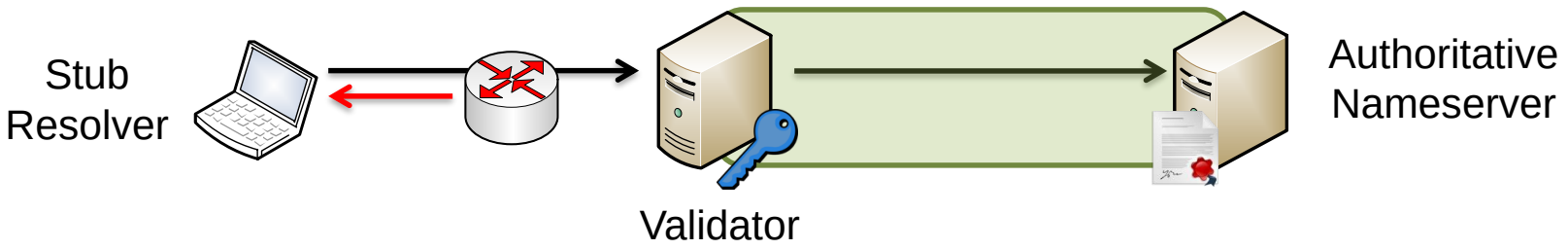
Key Ceremony



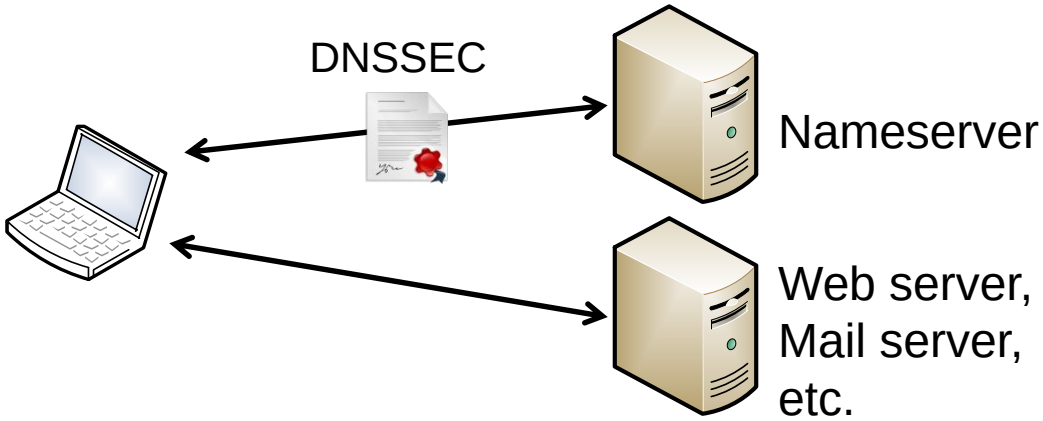


Limitations of DNSSEC

- No privacy: DNSSEC messages sent in cleartext
- Encryption is not supported
  - Enabling encryption would not help in all cases
  - e.g. encrypted query to 69.171.239.12 (a.ns.facebook.com)
- DNSSEC protects from misdirection
- But not from censorship by dropping messages
- DNSSEC protects between server and validating resolver
  - Path between end host and validator must be trusted



Limitations of DNSSEC (2)

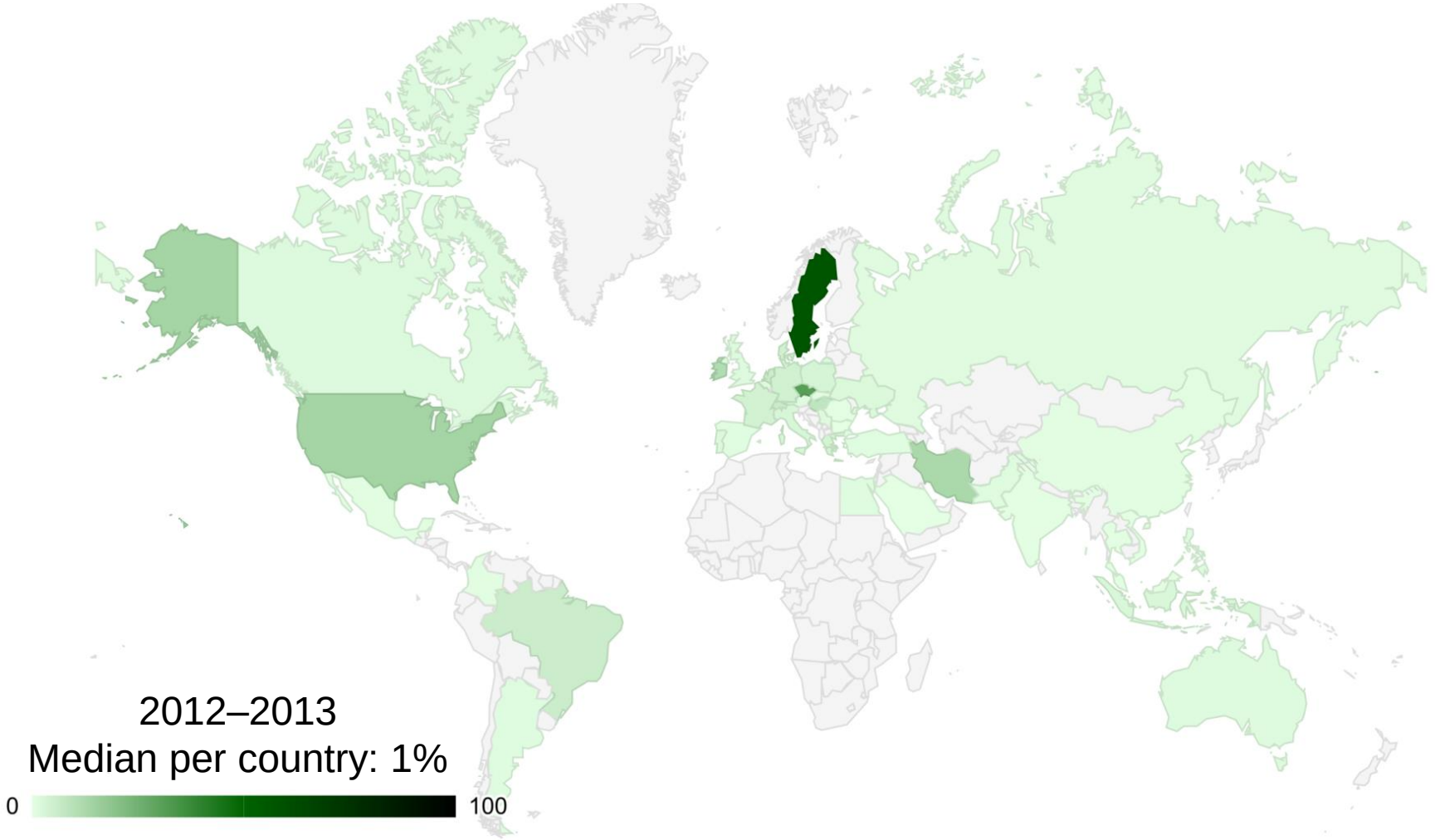


- DNSSEC secures the name resolution only
- Application traffic must be protected by other means
  - e.g. web with HTTPS
  - e.g. email with S/MIME
- DNSSEC complements but does not replace cryptographic protocols like TLS or SSH

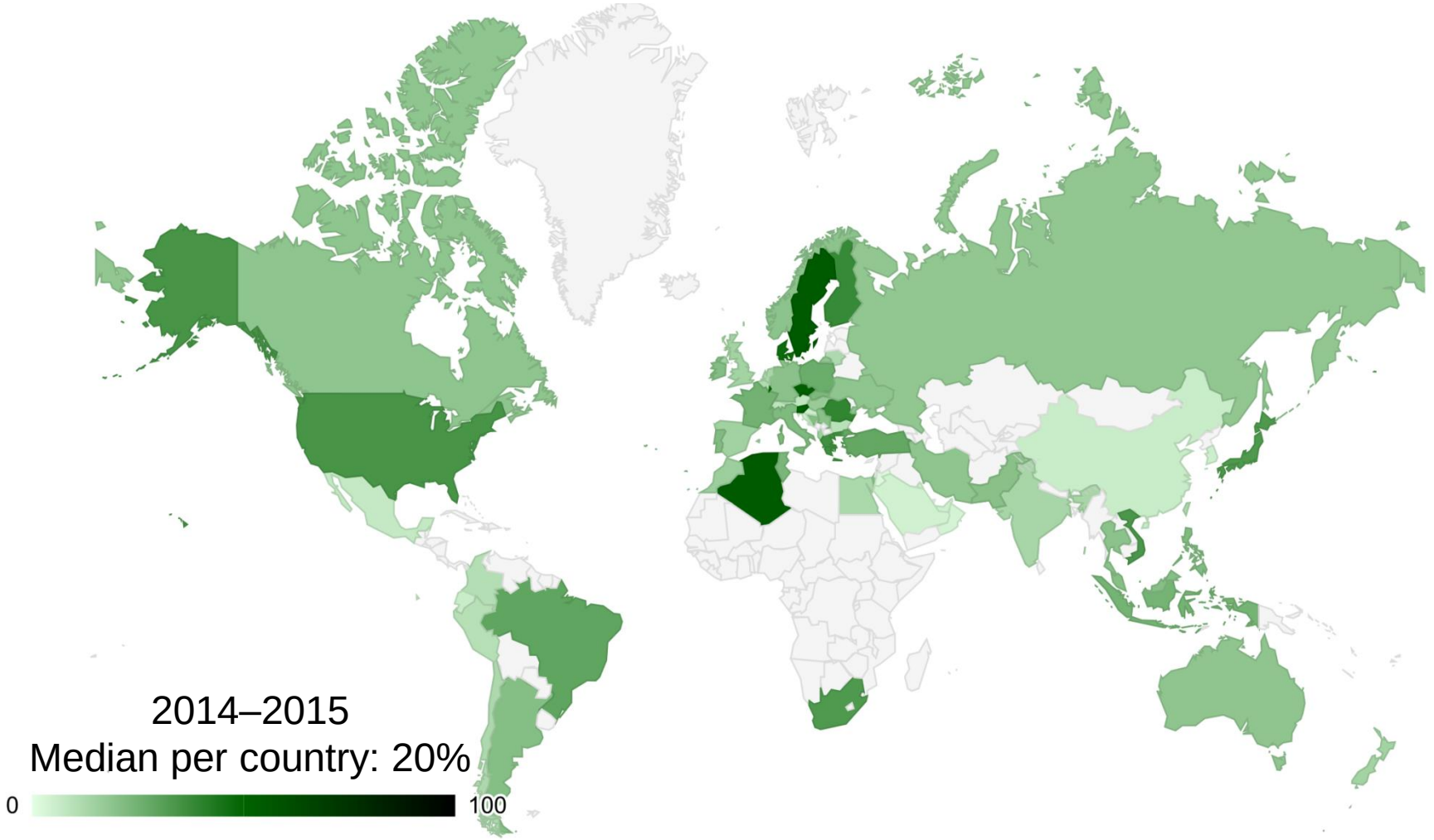
Use Case for DNSSEC: DANE

- Applications can rely on DNSSEC to retrieve securely certificates or public keys
- „DNS-based Authentication of Named Entities“ (DANE)
- Example: TLS certificate for HTTPS web servers
  - Any trusted certificate authority (CA) can issue X.509 certificates
  - Constraint allowed CAs with a TLSA record in DNSSEC domain
- Example: TLS certificate for mail servers
  - There is no secure way to determine whether a mail server supports TLS encryption on SMTP connections
  - Attacker can block TLS and force a downgrade to insecure SMTP
  - Set TLSA record on DNSSEC domain to indicate TLS support

DNSSEC Validation per Country

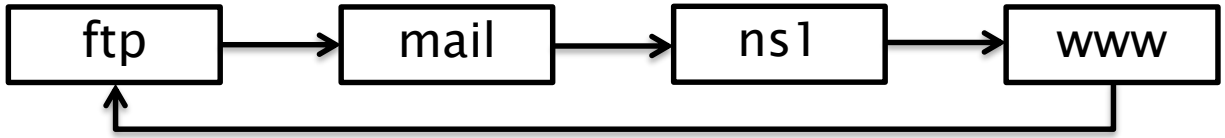


DNSSEC Validation per Country

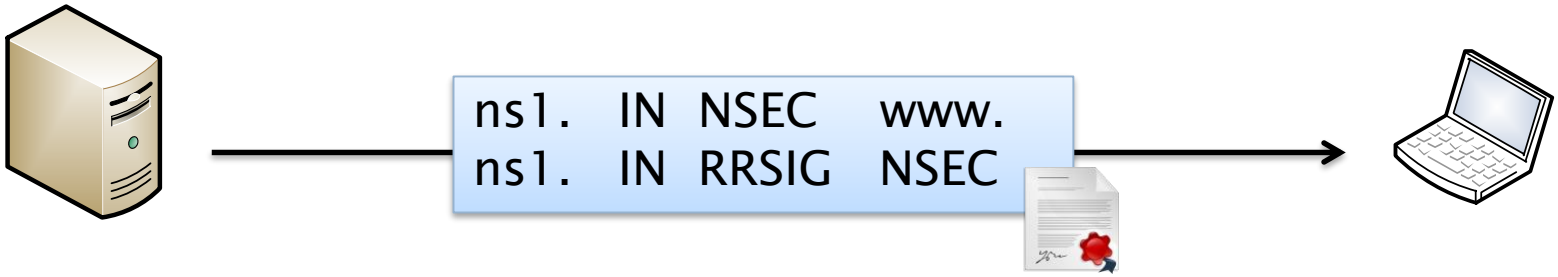


Authenticated Denial of Existence: NSEC

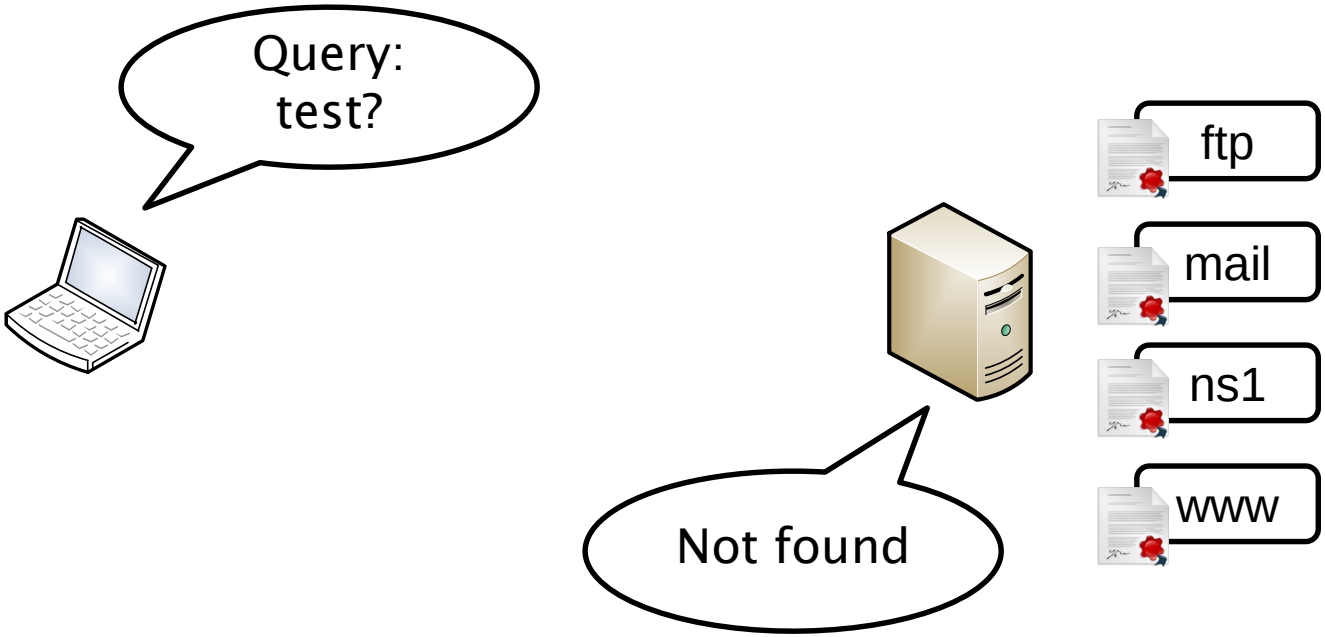
- DNSSEC signs resource records, not responses
- Negative responses („name not found“) have no records
- Chain existing domain names alphabetically to a ring



- NSEC record proves non-existence of domain name
  - Verify NSEC record with corresponding RRSIG record

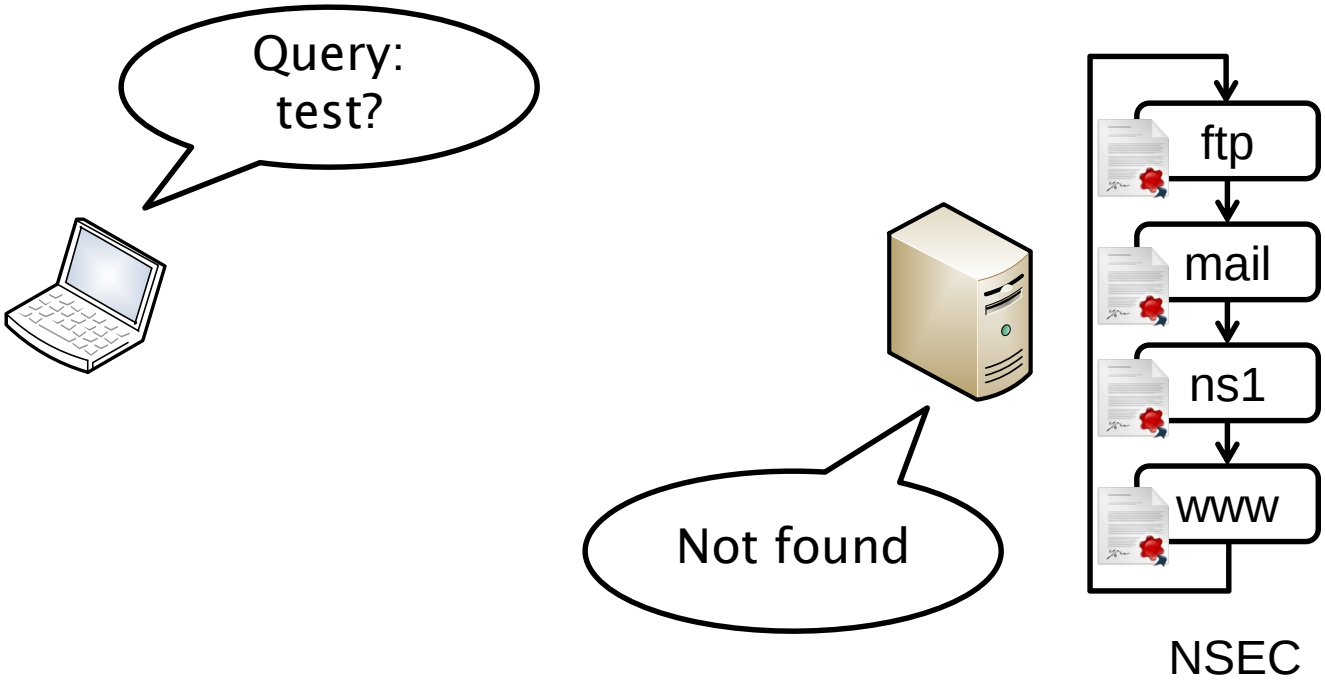


Negative Responses with NSEC



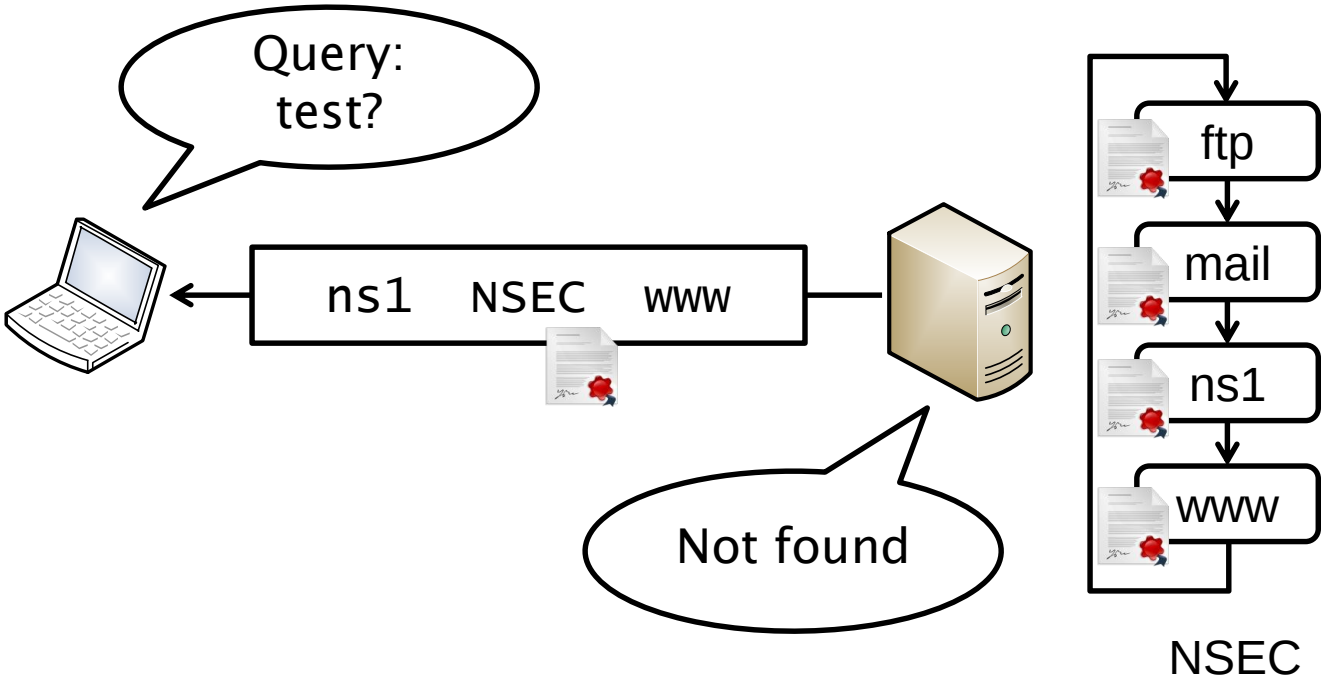
- Query name: test

Negative Responses with NSEC



- Query name: test

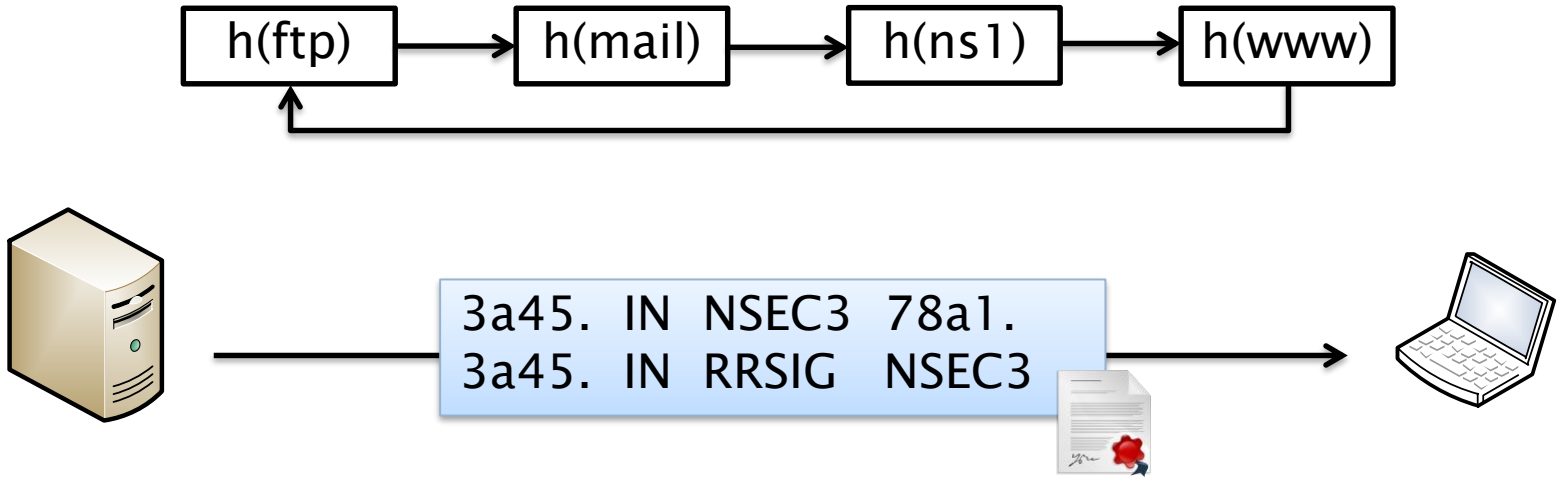
Negative Responses with NSEC



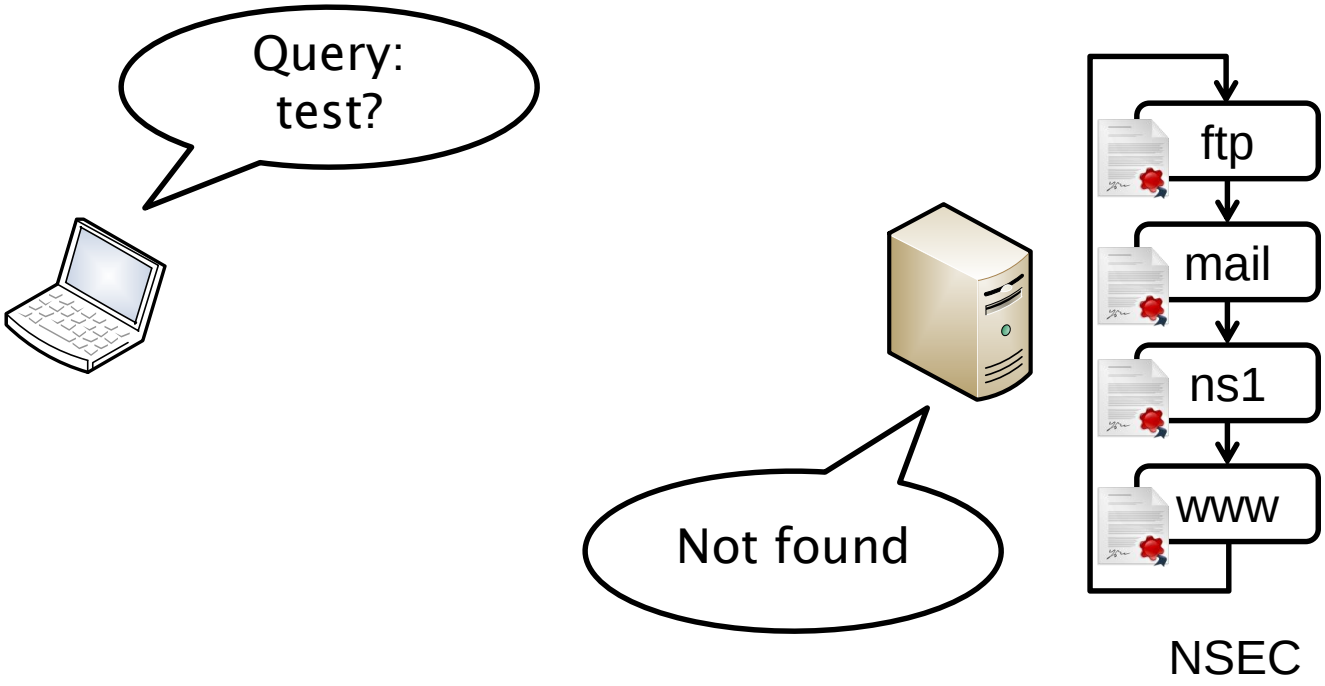
- Query name: test
- There is no ‘test’ between ‘ns1’ and ‘www’

Hashed Authenticated Denial of Existence: NSEC3

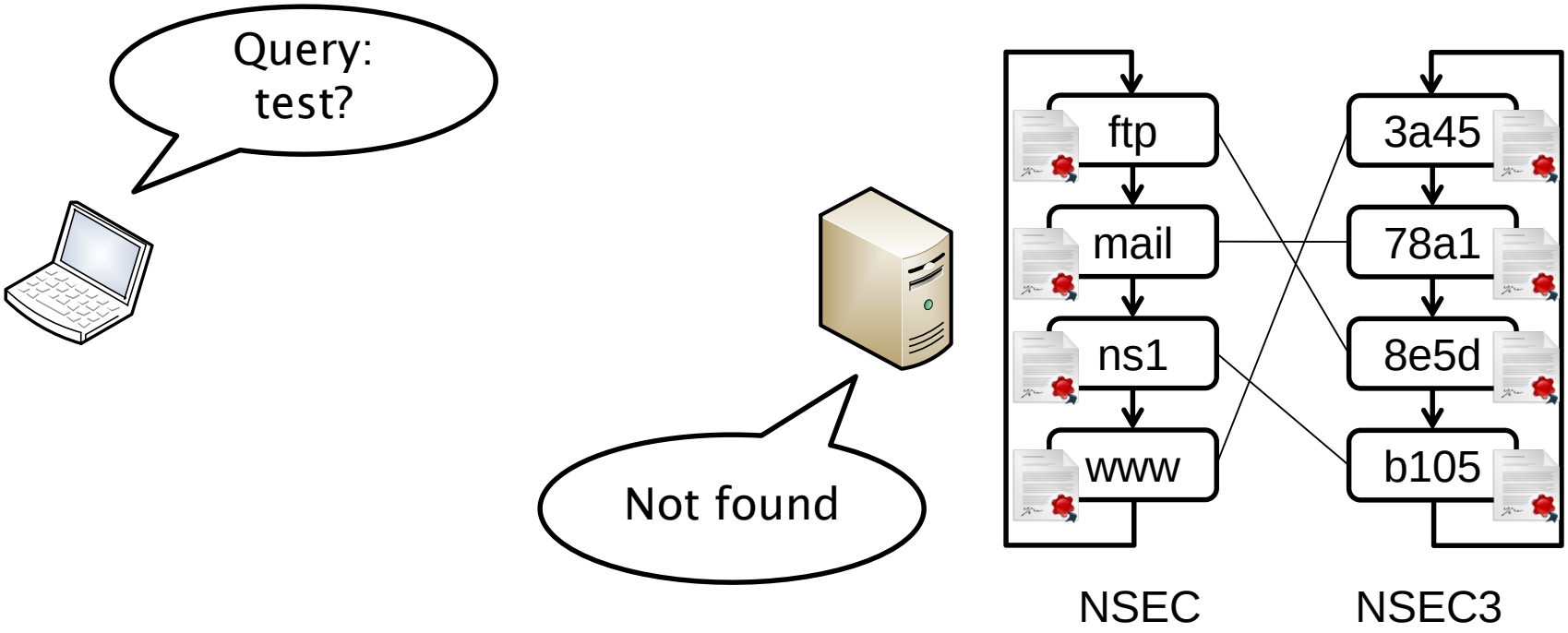
- NSEC records allow enumeration of all names
  - Anyone can create copy of whole domain database
- New requirement: hide names
- NSEC3: return hash values instead of cleartext names
- Chain hash values of names



Negative Responses with NSEC3

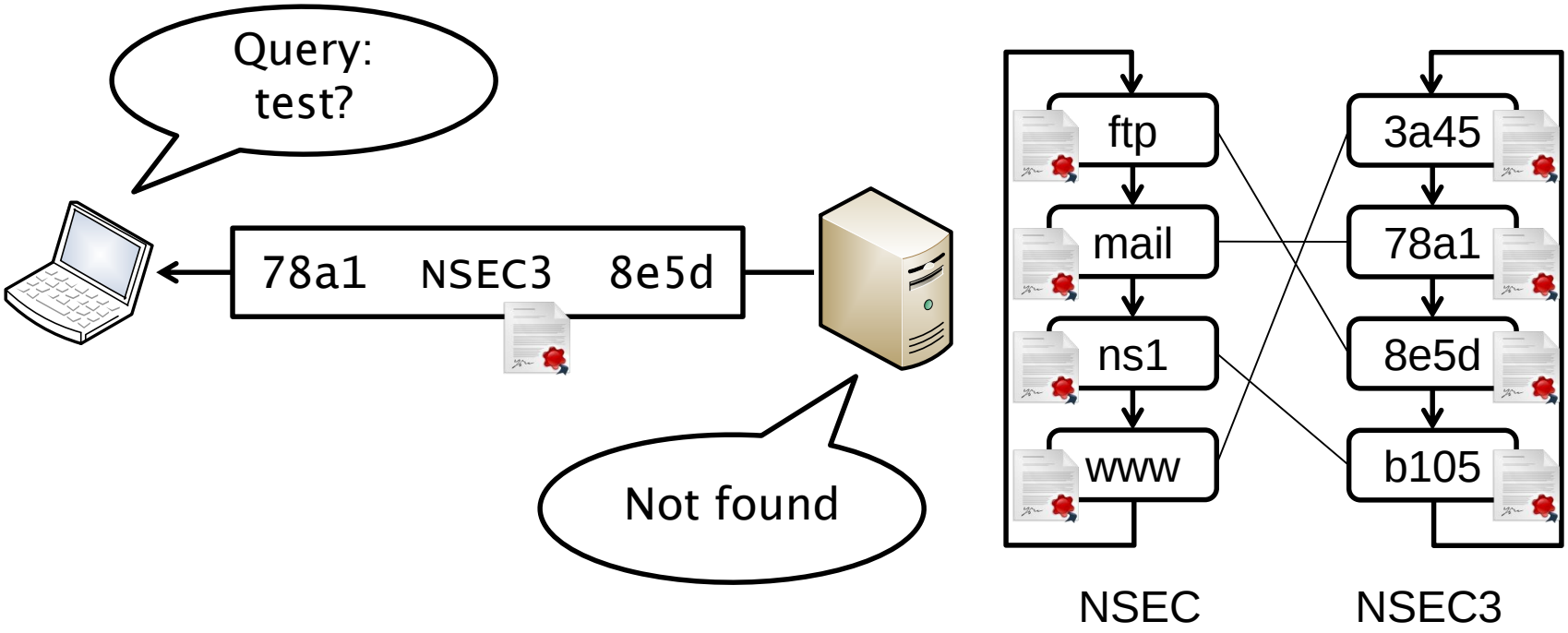


Negative Responses with NSEC3



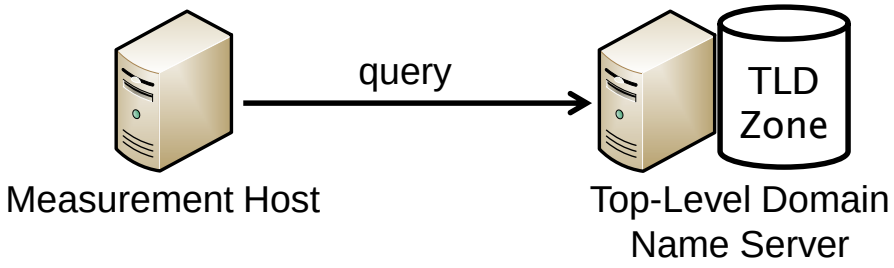
- Hash query name: h(test) ⇨ 810a

Negative Responses with NSEC3



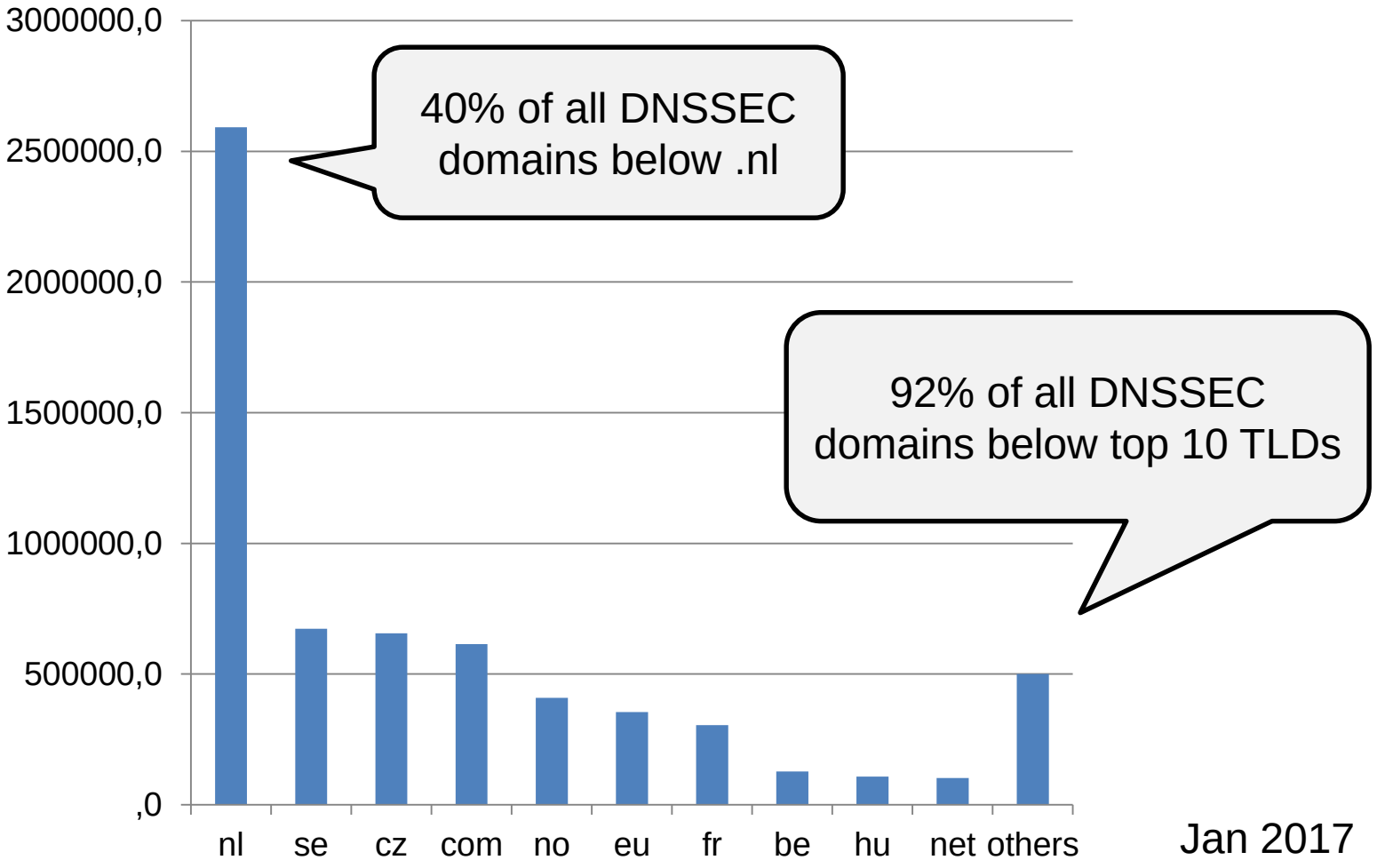
- Hash query name: h(test) ⇨ 810a
- There is no h(test) between 78a1 and 8e5d

Zone Enumeration

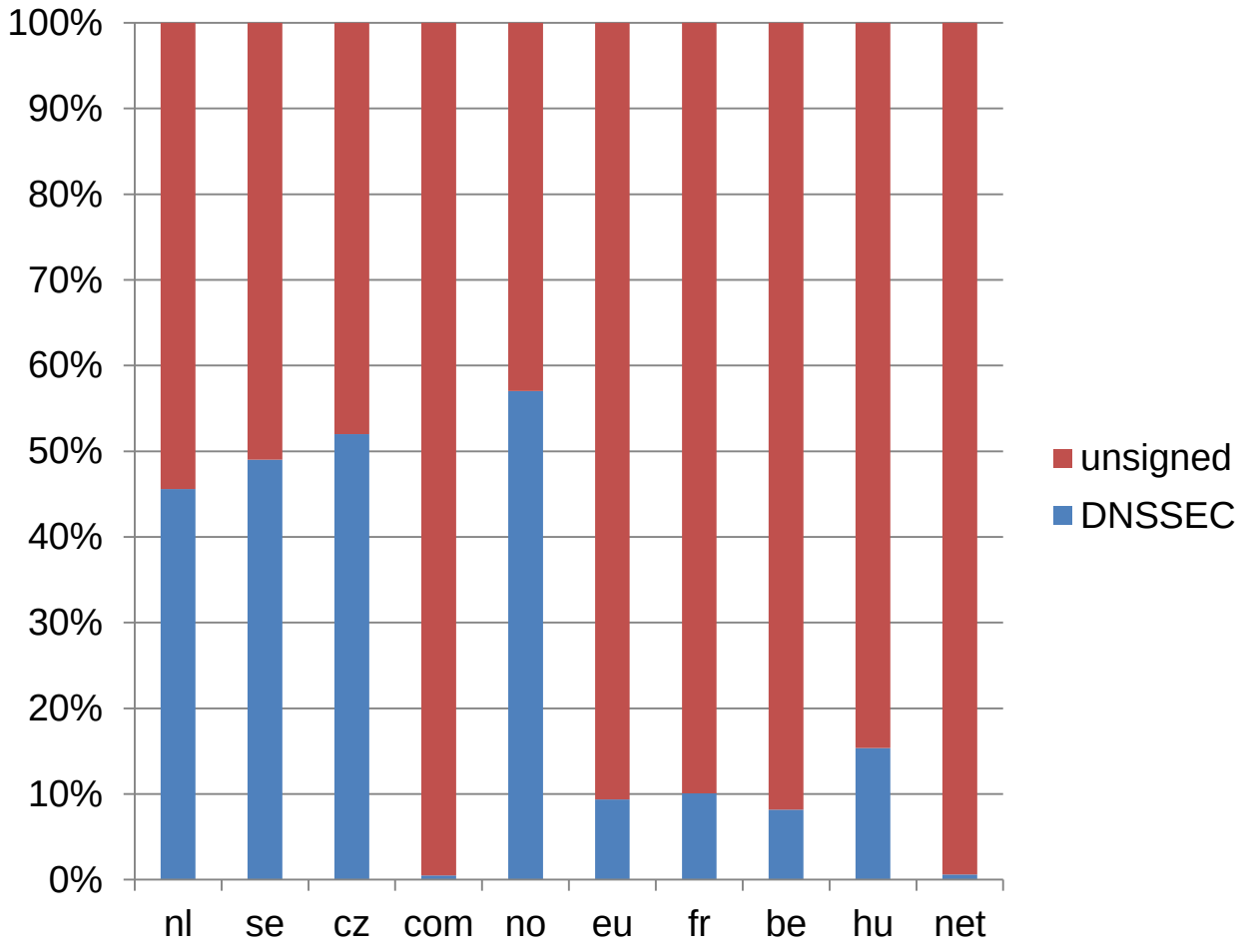


- Send queries for non-existing domain names to TLDs
  - Keep track of NSEC/NSEC3 records already known
  - 1 network query to retrieve 1 new NSEC/NSEC3 record
- NSEC zone enumeration yields domain names
- NSEC3 zone enumeration yields hash values
  - We don‘t see the domain name but can count them for statistics

DNSSEC Domains

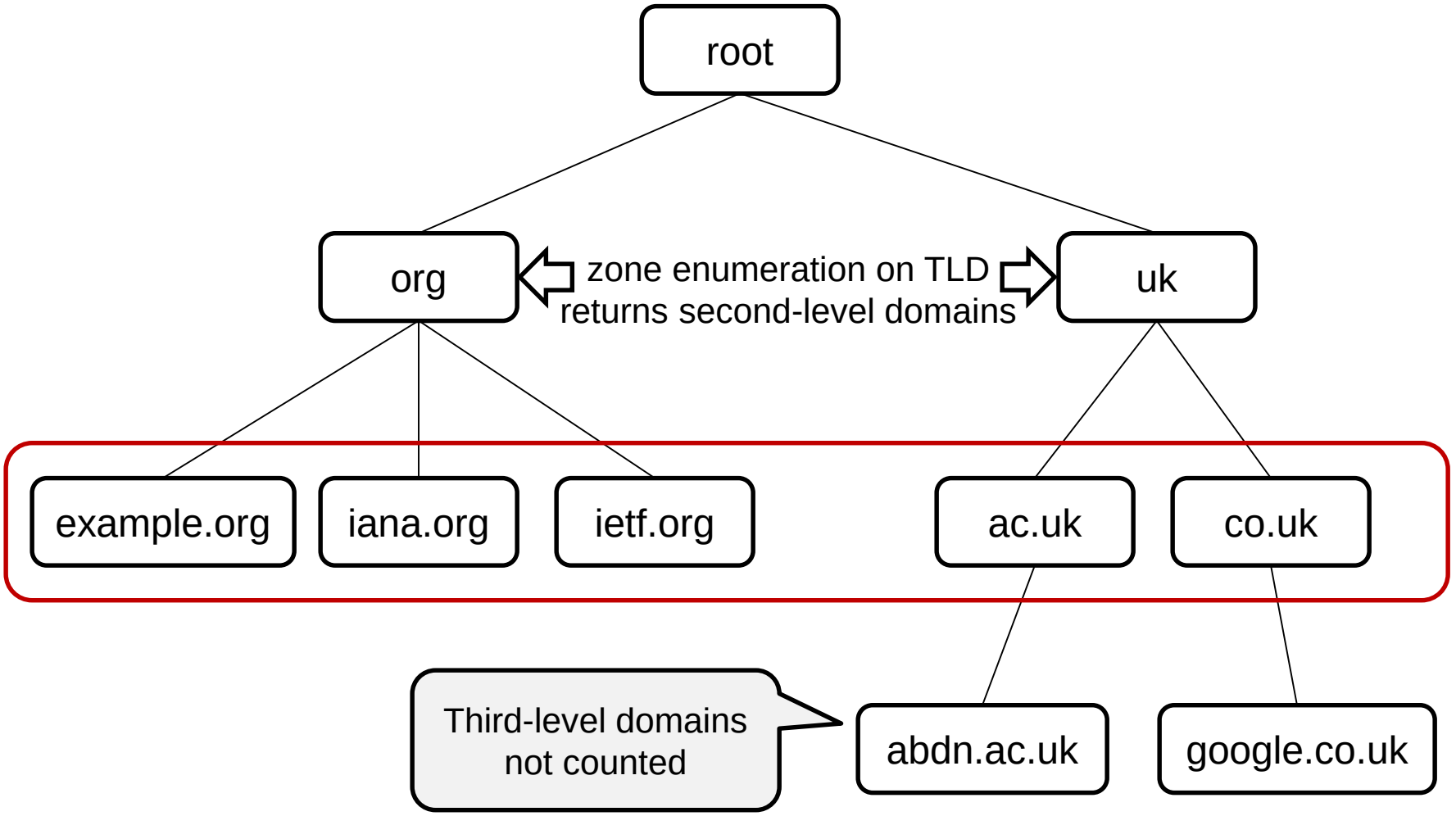


DNSSEC vs. unsigned Domains

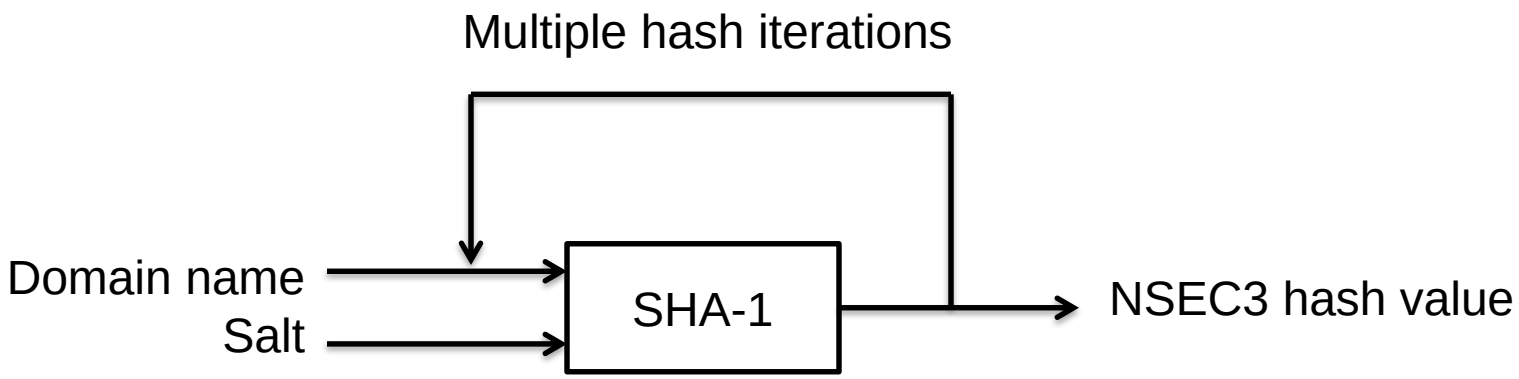


Jan 2017

Figure includes only Second-Level Domains



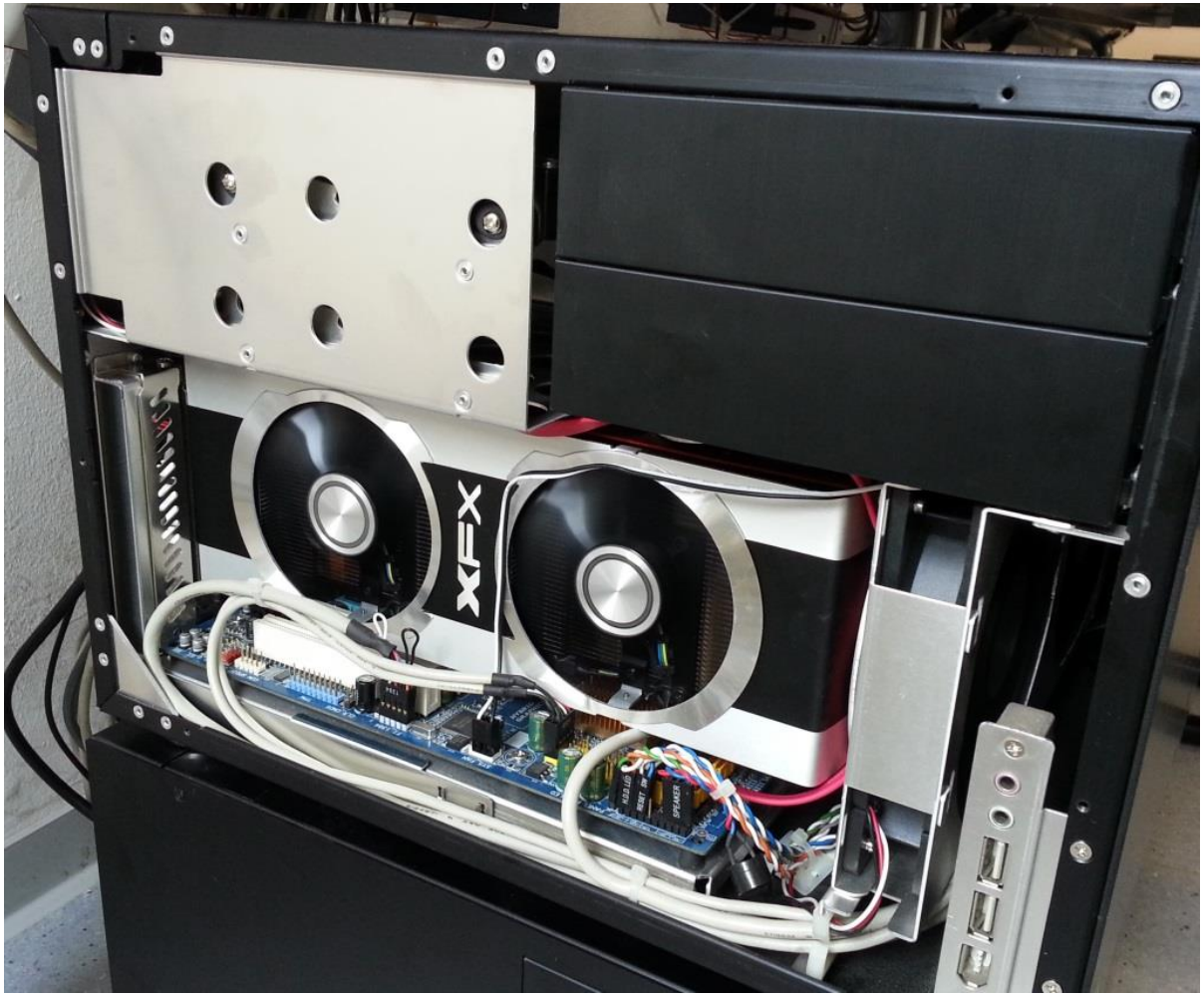
NSEC3 Hash Function: Iterated SHA-1 Hashing



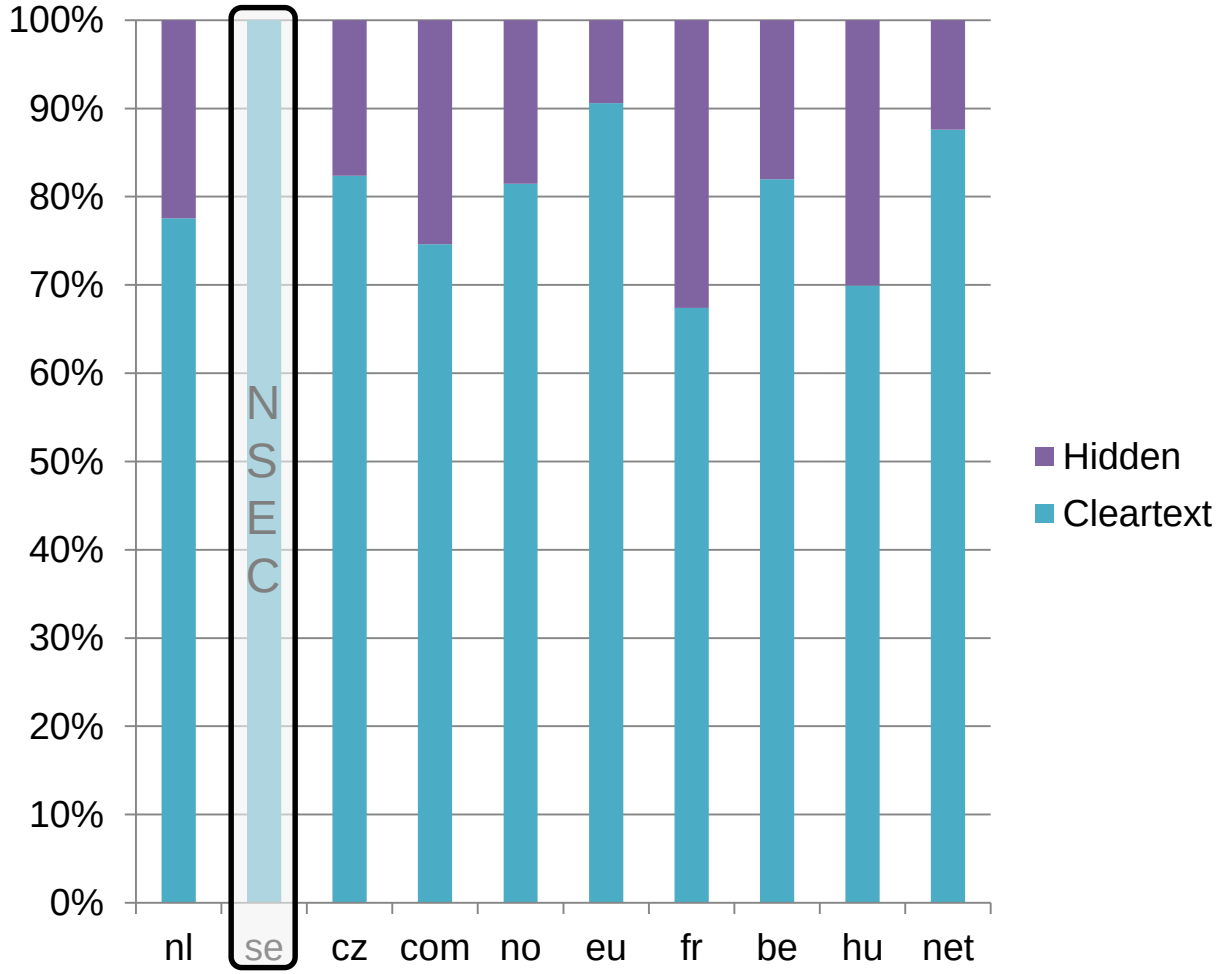
Hash Breaking Methods

- Brute-force attack
  - Exhaustive search (aaa, aab, aac, …)
  - Feasible only for short domain names
- Dictionary attack
  - Read candidate names from file
  - Generate additional candidates by recombining words
  - Very efficient and effective
- Markov attack
  - Derive candidate names from language model
  - Markov chain-based: what‘s the probability for certain character combinations in a given language?

Attack NSEC3 with GPU Computing



Cleartext Recovery Ratio



Jan 2017

What helps against Zone Enumeration?

- Broken NSEC/NSEC3 chain
  - Not practical: validation will fail on benign clients
- Frequent re-signing with new salt
  - Expensive: new signatures every few seconds/minutes
  - Beware: malicious attacker will increase query rate
- Online signing with NSEC3 or NSEC5
  - Expensive: new signature for each negative response
  - NSEC5 is similar to NSEC3, but replaces the hash function with an elliptic curve signing scheme

What helps not against Zone Enumeration?

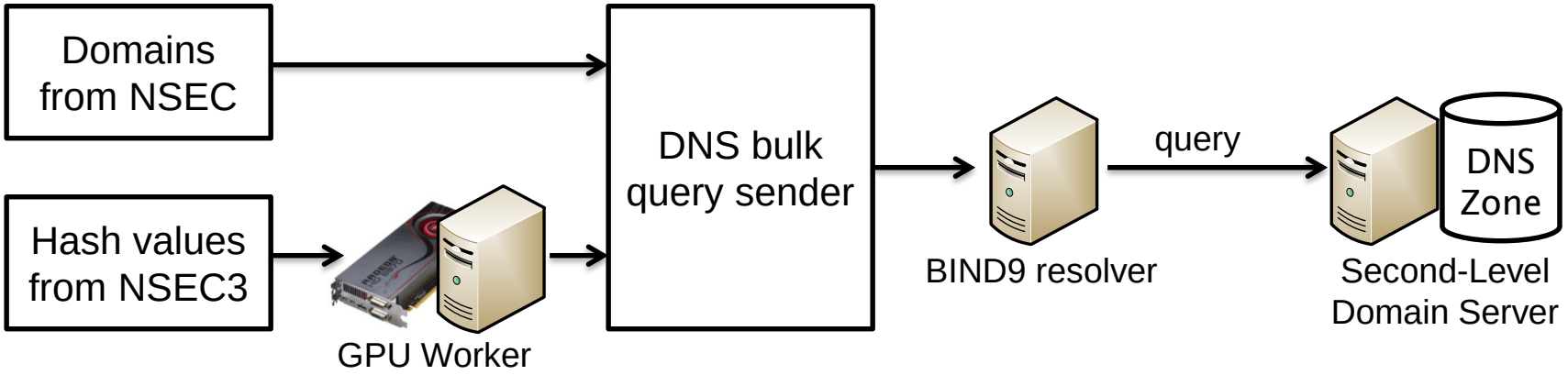
- Increase hash iteration count?
  - Slows down attack but not to a degree that helps

	TLD	NSEC or NSEC3?	DNSSEC Domains	Cleartext Recovery
22.	mx	NSEC3, opt-out, i=100	7,924	80%
132.	lat	NSEC3, opt-out, i=100	200	79%
187.	la	NSEC3, opt-out, i=150	105	96%
40.	name	NSEC3, opt-out, i=0	1,694	43%
71.	jp	NSEC3, opt-out, i=8	453	39%
112.	xn--3e0b707e 한국	NSEC3, opt-out, i=10	257	8%

Many iterations, high recovery

Few iterations, low recovery

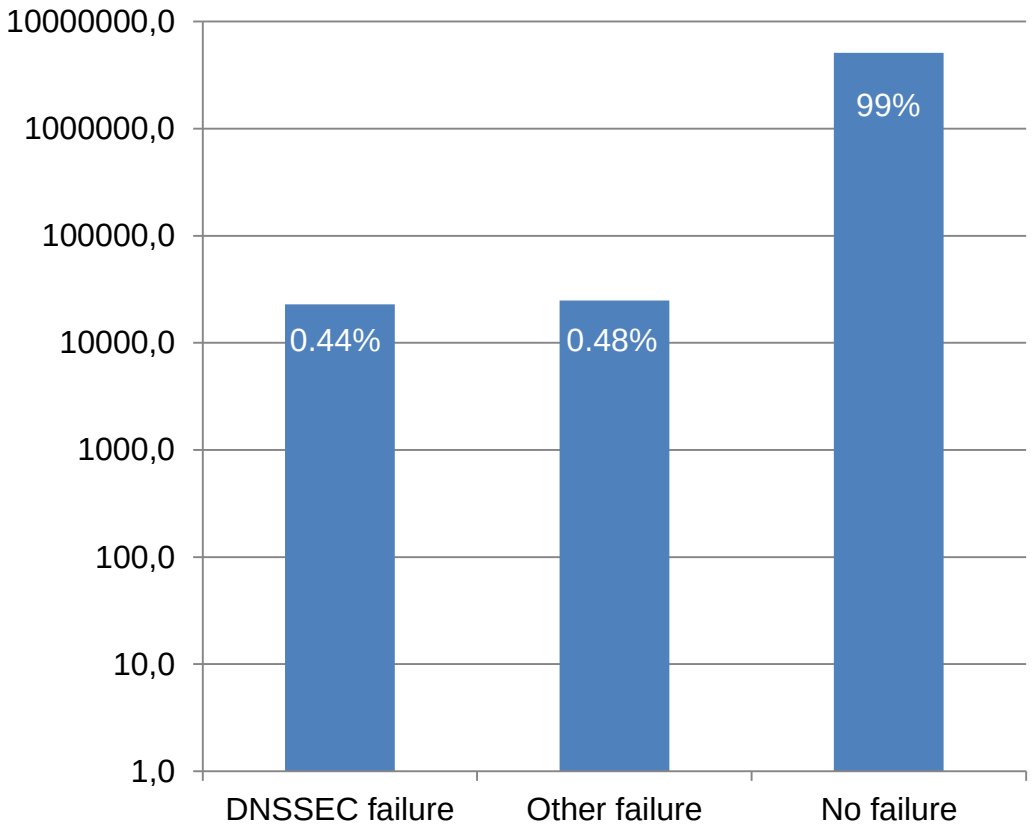
Analysis of Signed Second-Level Domains



- Query 5.1 million signed second-level domains for their DNSSEC configuration (DS and DNSKEY record sets)
- Check whether the domain validates correctly

Validation Failures

Result	Count
No DNSKEY (dangling DS)	19,386
No trusted DNSKEY (dangling DS)	1,216
No RRSIG for trusted DNSKEY	380
Signature expired	1,799
Signature ahead of time	1
Signature verify failure	49
Validation failure	**22,831**
Validation success	**5,092,022**



Jan 2017

Summary

- TSIG: point-to-point security
  - Shared secret must be known by both end points
- DNSSEC: end-to-end security
  - Signatures verified with public keys
- Public keys distributed within DNSSEC
  - Authentication chain for secure retrieval of public keys
  - Parent authorizes child public key via fingerprint (hash value)
- Negative responses authenticated indirectly
  - „bar NSEC foo“: ‚bar‘ and ‚foo‘ exist but nothing in between
  - NSEC: discloses cleartext names
  - NSEC3: returns hash values of existing names