

# Netzicherheit – Architekturen und Protokolle

## DNS Security



1. Einführung DNS
2. DNSsec





# Netzicherheit – Architekturen und Protokolle

## DNS Security

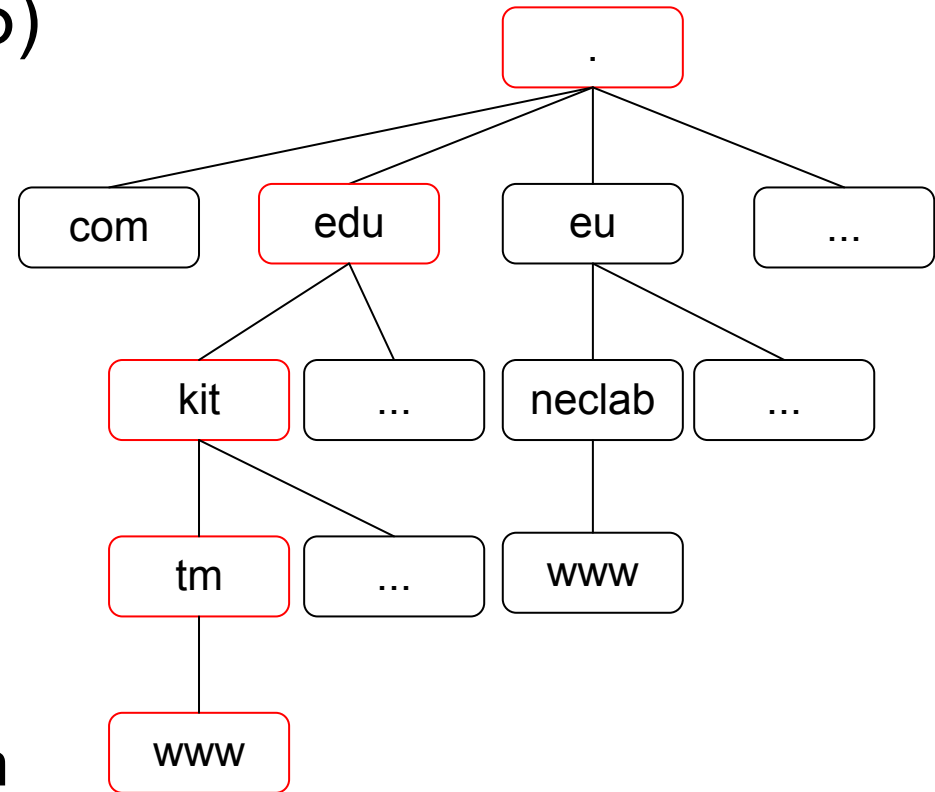


1. Einführung DNS
2. DNSsec



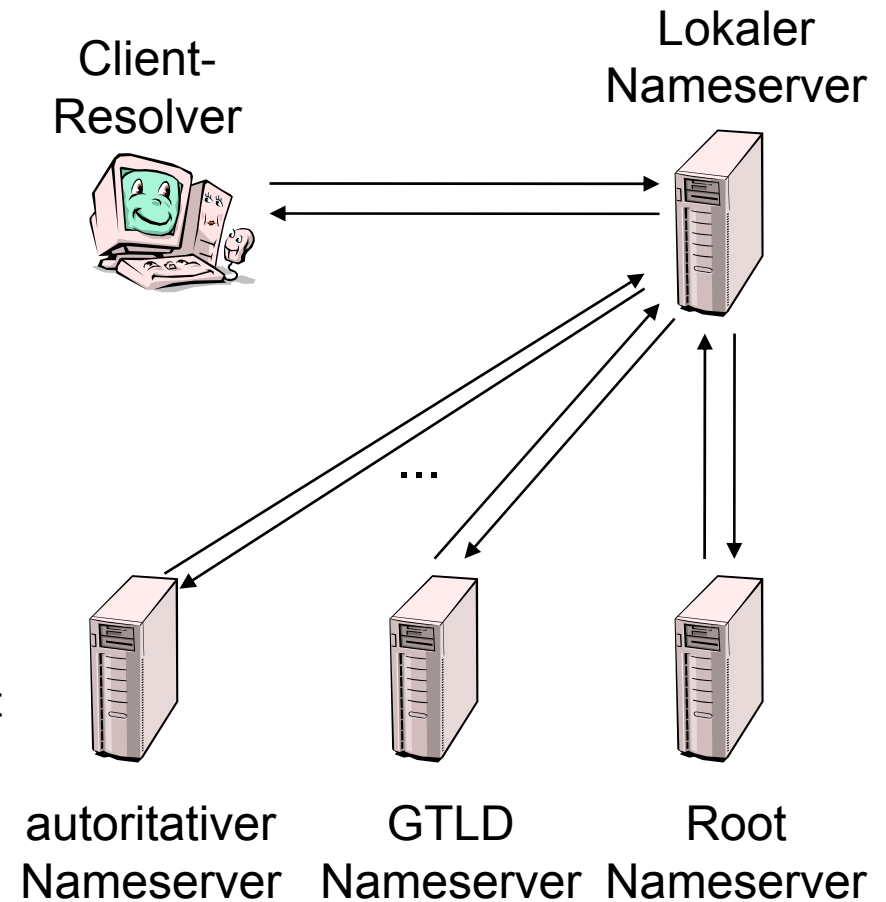
- **Domain Name System (DNS) (RFC 1034 / 1035)**
  - Namensauflösung von Domainnamen nach IP Adresse
  - Verteilter hierarchischer Verzeichnisdienst
  - Standard UDP Port 53 – Backup TCP Port 53
  - Maximale Paketgröße: 512 Byte
    - ▶ Einführung von EDNS (RFC 2671)
    - ▶ Variable UDP-Größe
    - ▶ Erweiterung um 16 neue Flags
    - ▶ DNSsec erfordert zwingend EDNS wegen neuem Flag und Größe der DNS Antwort
  - Modi:
    - ▶ Rekursive Namensauflösung
    - ▶ Iterative Namensauflösung

- Domain Names (RFC1035)
  - Max. Länge: 255 Zeichen
  - Delegation von rechts nach links
- Hierarchischer Aufbau in Form eines Baums
  - Erste Ebene: Top-Level-Domains (TLD)
  - Blätterbezeichner zwischen 1 und 63 Zeichen



Beispiel FQDN: `www.tm.kit.edu`

- **Rekursive** Namensauflösung
  1. Resolver (Client) stellt Anfrage an lokalen NS (z.B. des ISP)
    - NS Kennt die Abbildung Name → IP nicht
  2. Wählt einen Root-NS und leitet die Anfrage weiter
    - Dieser kennt die Antwort nicht und delegiert an GTLD (global TLD)
  3. Wählt einen GTLD-NS und wiederholt die Anfrage
    - Kennt die Antwort auch nicht und delegiert
  4. Wiederholung der Anfrage bis zum zuständigen (autoritativen) NS
    - Sendet die Abbildung als Antwort
  5. Lokaler NS sendet Antwort an Client
    - Antwort ist nicht autoritativ!



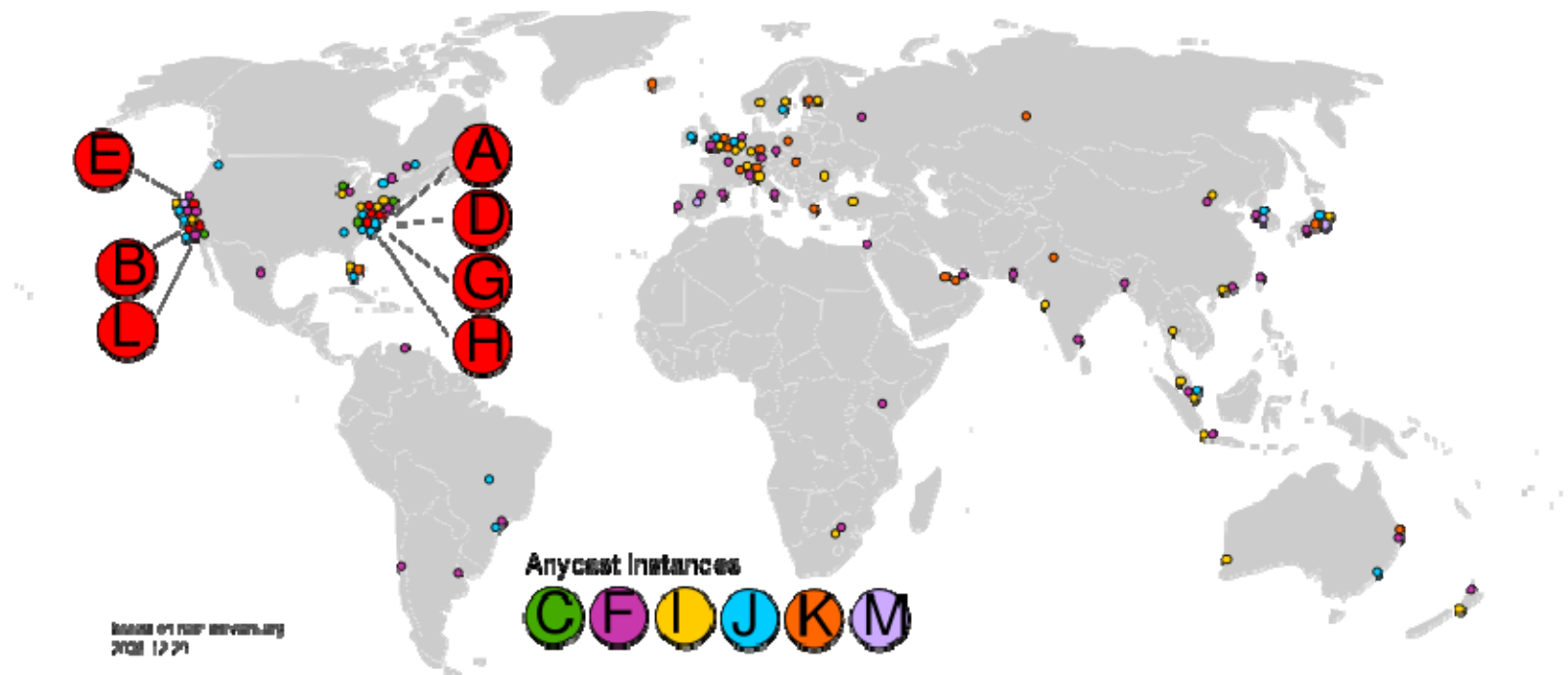


- Namensauflösung gesteuert durch Konfiguration des Resolvers
    - *Hosts*-Datei statisch konfiguriert
    - Standard Auflösungsreihenfolge: hosts, dns
  - **Pharming manipuliert die *hosts*-Datei**
    - Eintragen von statischen Abbildungen
    - Meist durch Virus oder Trojaner
    - Umleiten von Verkehr auf gefälschte Seite um
      - ▶ Login-Daten abzugreifen
      - ▶ Personen-bezogene Daten zu stehlen
- **Für den Benutzer absolut nicht erkennbar**



- **Resolver**
  - Client Software des DNS, das Anfragen stellt und Antworten entgegennimmt
  - Besitze eigene minimal Konfiguration
- **Zone**
  - Administrative Domäne eines Netzanbieters
- **Resource Record**
  - DB-Typ der Abbildung
  - Z.B. Name → IPv4
  - IPv6, MX, NS, SOA
- **Nameserver (NS)**
  - Server software
- **Autoritativer NS**
  - Löst die angefragte Abbildung auf
- **Delegation**
  - Verweis auf einen anderen NS, um die angefragte Abbildung aufzulösen
- **Rekursiver NS**
  - Ein NS, der alle Delegation bis zum autoritativen NS folgt

- 13 Root-Server weltweit verteilt
  - 7 zentralistisch
  - 6 verteilt auf Anycast-Basis
- Verteilung





- **Neue Daten**
  - Aufspielen neuer Abbildungen
- **NS-Replizierung**
  - Master-Slave Ansatz in einer Zone
  - Lastverteilung und Ausfallsicherheit
- **Dynamische Updates**
  - Änderungen per DNS-Request
- **TSIG Resource Record**
  - Meta-RR
  - Integritätssicherung der DNS Daten mittels MD5
  - Vorverteiltes Geheimnis
  - Keine Verschlüsselung
- **Nachteil**
  - Schlechte Skalierbarkeit
  - Nur in kleinen isolierten Netzen sinnvoll einsetzbar

- IP Adressen
  - Fälschen der Quell IP-Adresse einfach
- UDP Port Nummer
  - Server: 53/udp
  - Client: 53/udp, fixed config, fixed on random init
- Query ID
  - Eindeutige ID, um Anfrage und Antwort zuzuordnen
  - Häufig: 1-er Inkrement für jede Anfrage

Ver	hlen	protocol	Packet length										
identification			flg	offset									
TTL		protocol	Header checksum										
Source IP address													
Destination IP address													
Source. Port				Destination Port									
UDP length				UDP cchecksum									
Query ID			Q R	OP code	AA	TC	RD	RA	Z	rcode			
Question count			Answer Count										
Authority count			Add Record Count										
DNS question or DNS answer data													

IP header

UDP header

DNS Data

- Anfrage nach [www.neclab.eu](http://www.neclab.eu)
  - Lokaler NS: kyoto.neclab.eu
  - Gesucht IPv4 Adresse
- Beispiel: Nachricht von Kyoto an TLD NS
  - Query Count = 1 zeigt an, dass im Datenfeld ein Anfrage enthalten ist
  - QR = 0: Anfrage
  - OP = 0: Standardanfrage
  - RD = 1: Rekursive Auflösung gewünscht

Ver	hlen	protocol	Packet length					
identification			flg	offset				
TTL		protocol	Header checksum					
195.37.70.24 (kyoto.neclab.eu)								
192.112.36.4 (g.root-servers.net)								
1234			53					
UDP length			UDP ckecksum					
54321			0	0	ATT AC	1R A	Z	rcode
Question count = 1			Answer Count = 0					
Authority count = 0			Add Record Count = 0					
Qu: Wie ist die IPv4 Adresse von <a href="http://www.neclab.eu">www.neclab.eu</a> ?								

IP header

UDP header

DNS Data

- Antwort des TLD
  - QR = 1: Antwort
  - AA = 0: nicht authorativ
  - RA = 0: Rekursion nicht unterstützt
- Delegation
  - Liste von NS für nächsten Rekursionsschritt (Authority Count = 2)
  - List von A-Records für diese NS (Add Record Count = 2)
- Anfrage-Antwort Verknüpfung
  - NS hält Liste von ausstehenden Anfragen (Query-ID)
  - Ignorieren von Antworten ohne Query-ID in Liste

Ver	hlen	protocol	Packet length					
identification			flg	offset				
TTL		protocol	Header checksum					
192.112.36.4 (g.root-servers.net)								
195.37.70.24 (kyoto.neclab.eu)								
53			1234					
UDP length			UDP cchecksum					
54321			1	0	0	0	Z	ok
Question count = 1			Answer Count					
Authority count = 2			Add Record Count = 2					
Qu: Wie ist die IPv4 Adresse von <a href="http://www.neclab.eu">www.neclab.eu</a> ?								
Au: neclab.eu NS = x.nic.eu								
Au: neclab.eu NS = l.eu.dns.be								
Ad: x.nic.eu A = 194.0.1.19								
Ad: l.eu.dns.be A = 193.2.221.60								

IP header

UDP header

DNS Data

Authority

Glue



- Antwort von **kyoto.neclab.eu**
  - QR = 1: Antwort
  - AA = 1: autoritativ
  - RA = 0: Rekursion nicht unterstützt
- Canonical Name
  - Alias oder alternativer Name
  - Antwort enthält keine Abbildung zu IPv4 Adresse
  - Weitere Namensauflösung notwendig

Ver	hlen	protocol	Packet length					
identification			flg	offset				
TTL	protocol		Header checksum					
192.112.36.4 (g.root-servers.net)								
195.37.70.24 (kyoto.neclab.eu)								
53			1234					
UDP length			UDP cchecksum					
54322			1	0	1	0	Z	ok
Question count = 1			Answer Count = 1					
Authority count = 2			Add Record Count = 2					
Qu: Wie ist die IPv4 Adresse von <a href="http://www.neclab.eu">www.neclab.eu</a> ?								
An: <a href="http://www.neclab.eu">www.neclab.eu</a> CNAME = www.netlab.nec.de								
Au: nec.de NS = a.nic.de								
Au: nec.de NS = l.de.net								
Ad: a.nic.de A = 194.0.0.53								
Ad: l.de.net A = 77.67.63.105								

IP header

UDP header

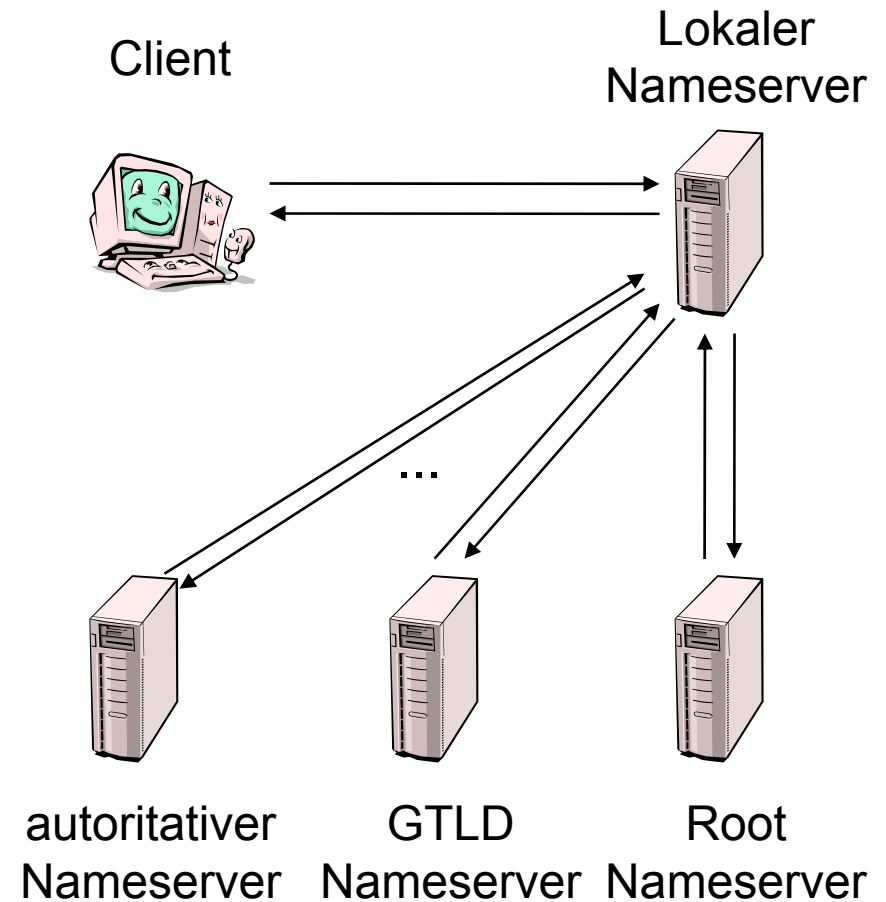
DNS Data

Answer

Authority

Glue

- Was muss man tun, damit der lokale NS eine gefälscht/gespoofte DNS-Antwort annimmt?



- **DNS cache poisoning**

## Angriff

- Eintrag gefälschter Abbildungen in den DNS Cache
- Übernahme von Zonen möglich

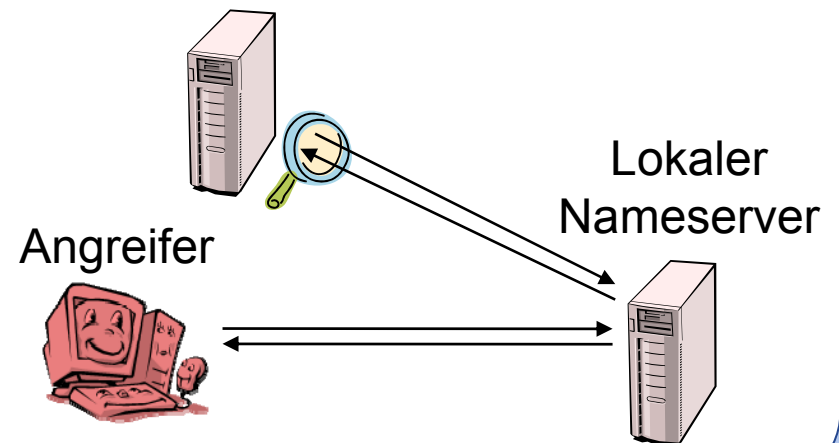
- **Kein Phishing!**

- Resultat ähnlich
- Phishing kann von Benutzer erkannt werden

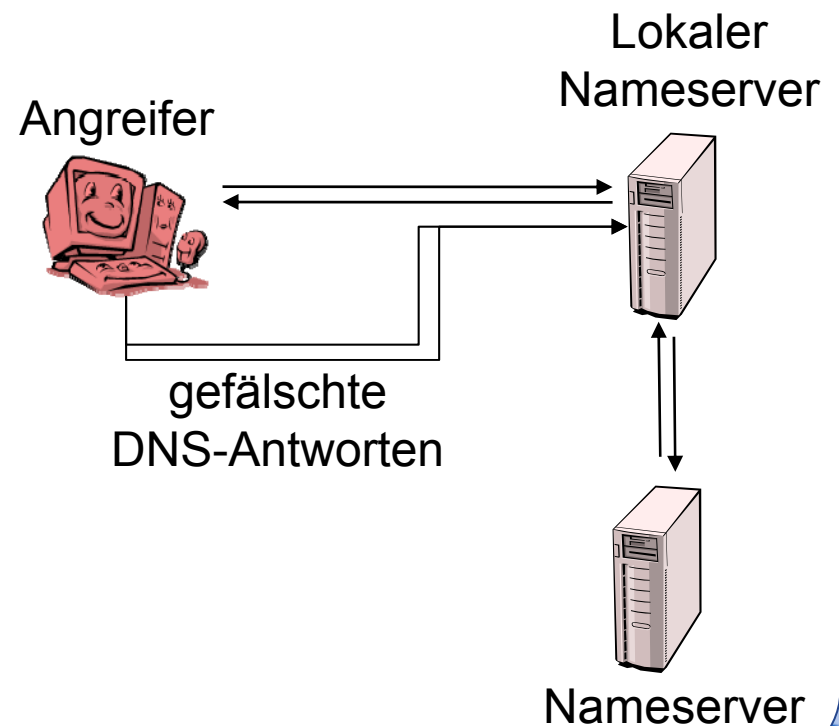
- **Vorbereitung**

- DNS Anfrage eines Namens, für den ein kontrollierter NS autoritativ ist
- Auslesen, des UDP Source Ports und des aktuellen Query-ID-Werts

Nameserver  
unter Kontrolle des Angreifers



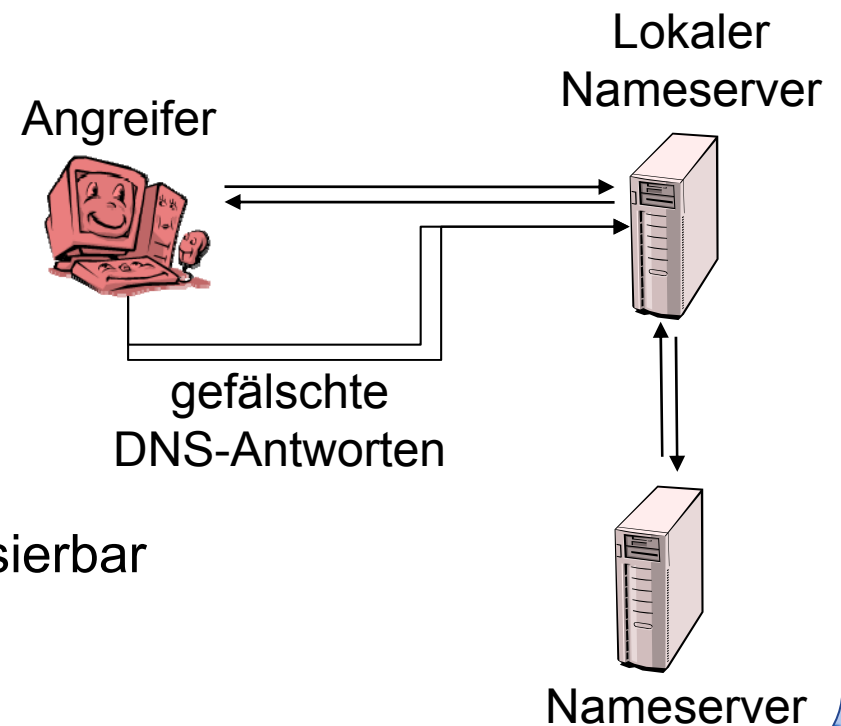
- Annahme
  - UDP source port statisch
  - Query-ID wird für jede Anfrage inkrementiert
  - Abbildung nicht im DNS-cache
- Angriff auf lokalen NS
  - DNS-Anfrage nach [www.trustedbank.com](http://www.trustedbank.com)
  - Senden von DNS-Antworten mittels IP-Spoofing und verschiedenen Query-Ids
    - ▶ Enthalten gefälschte Abbildung
  - Angriff erfolgreich, wenn
    - ▶ die richtige Query-ID erraten wird
    - ▶ Antwort den lokalen NS vor der richtigen DNS Antwort erreicht





- Erweiterung des einfachen Angriffs

- Setup eines NS für trustedbank.com
- DNS-Anfrage für einen Namen, der nicht im Cache ist
  - ▶ Vorschläge?
- Fälschen der Root- oder GTLD-Antwort
  - ▶ Gefälschte Glue-Einträge
- Übernahme der gesamten Zone
- Erfolgswahrscheinlichkeit pro Zone gering, aber einfach automatisierbar



- Keine echte Lösung des Problems, aber schnell ausrollbar
  - Zufällige Query-ID
    - ▶ Beschränkt auf 16-bit wegen Kompatibilität
  - Zufällige UDP-Portnummer
    - ▶ Speichern der verwendeten Portnummer mit DNS-Anfrage
    - ▶ Beispiel: Microsoft DNS Server reserviert 2500 Portnummern
- Echte Lösung: DNSsec
  - Problem: Umstellen der Infrastruktur erst begonnen

# Netzicherheit – Architekturen und Protokolle

## DNS Security



1. Einführung DNS
2. DNSsec



- Formulieren Sie Designkriterien für ein sicheres Namensauflösungssystem
- Wie können diese technisch umgesetzt werden?
  - Welche Vorteile bzw. Nachteile haben die unterschiedlichen Ansätze?



- Schwachstellen des DNS bekannt seit 90er
  - Steven Bellovin: *Using the Domain Name System for System Break-ins*, veröffentlicht 1995
- IETF WG: DNS Security
  - 1997: RFC 2065 veröffentlicht
  - 1999: Überarbeitung als RFC 2535 standardisiert
  - 2001: Skalierungsprobleme entdeckt
    - ▶ Schlüsselerneuerung erfordert 6 Nachrichten zwischen Zone und Vaterzone
    - ▶ Schlüsselerneuerung einer TLD-Zone hätte Millionen Update-Nachrichten zu allen Kindzonen erfordert
  - 2005: RFCs 4033 – 4035 veröffentlicht
  - 25.01.2010: L-Rootserver signiert (DURZ)
  - Mai 2010: Alle Rootserver DNSSec fähig
  - 01.07.2010: Umstellung auf echten Schlüssel

# DNS KEY (DNSKEY) RR

- Beispiel aus RFC 4034

Besitzer	TTL	Class	Type	ZSK	Protokol	Alg.
<i>dskey.example.com</i>	<i>86400</i>	<i>IN</i>	<i>DNSKEY</i>	<i>256</i>	<i>3</i>	<i>5 (</i>

*AQOeiiR0GOMYkDshWoSKz9Xz  
fwJr1AYtsmx3TGkJaNXVbfi/  
2pHm822aJ5iI9BMzNXxeYCMZ  
DRD99WYwYqUSdjMmmAphXdvx  
Öffentlicher  
Schlüssel  
egYd/M5+X7OrzKBaMbCVdFLU  
Uh6DhweJBjEVv5f2wwjM9Xzc  
nOf+EPbtG9DMBmADjFDc2w/r IjwvFw== ) ;*

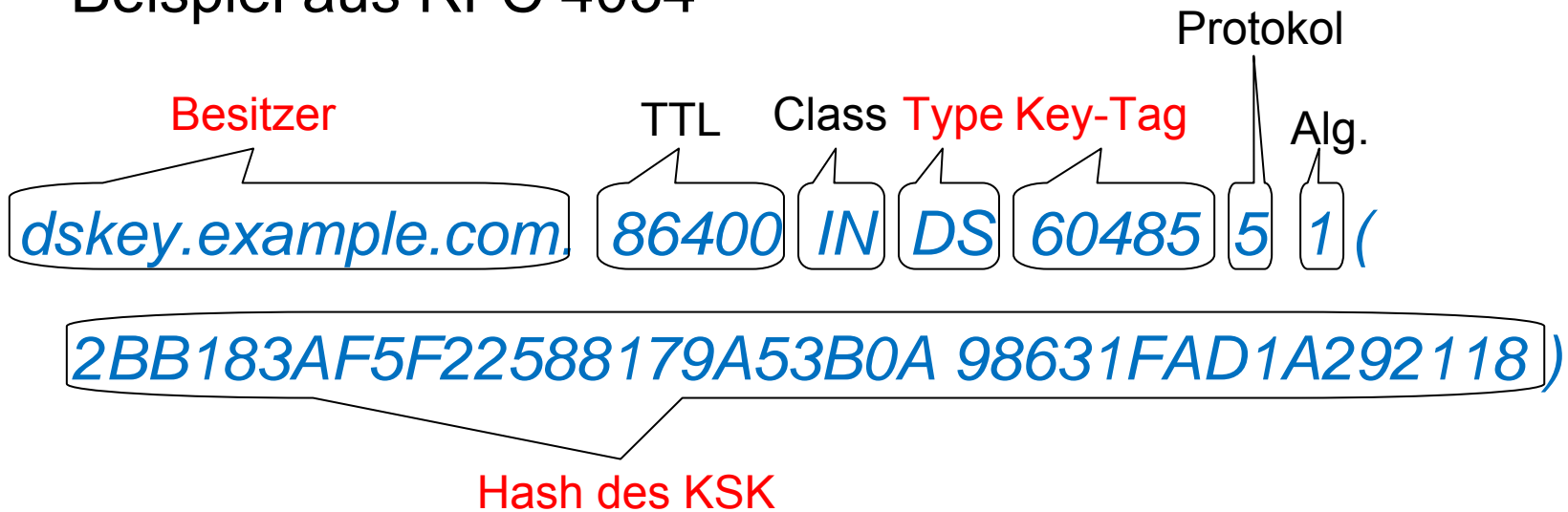
*key id = 60485*

Key-Tag

- Zone Signing Key (ZSK)
  - Öffentlicher Schlüssel abgelegt in Zonendatei
  - Signierung der RRs der Zone
  - Signiert durch KSK der Zone
  - Flag: 256
- Key Signing Key (KSK)
  - Öffentlicher Schlüssel abgelegt in Zonendatei
  - Signierung des ZSK der Zone
  - Signiert durch den ZSK der Vaterzone
  - Flag: 257
- Trennung in zwei Schlüssel erleichtert Schlüsselerneuerung

# Delegation Signer (DS) RR

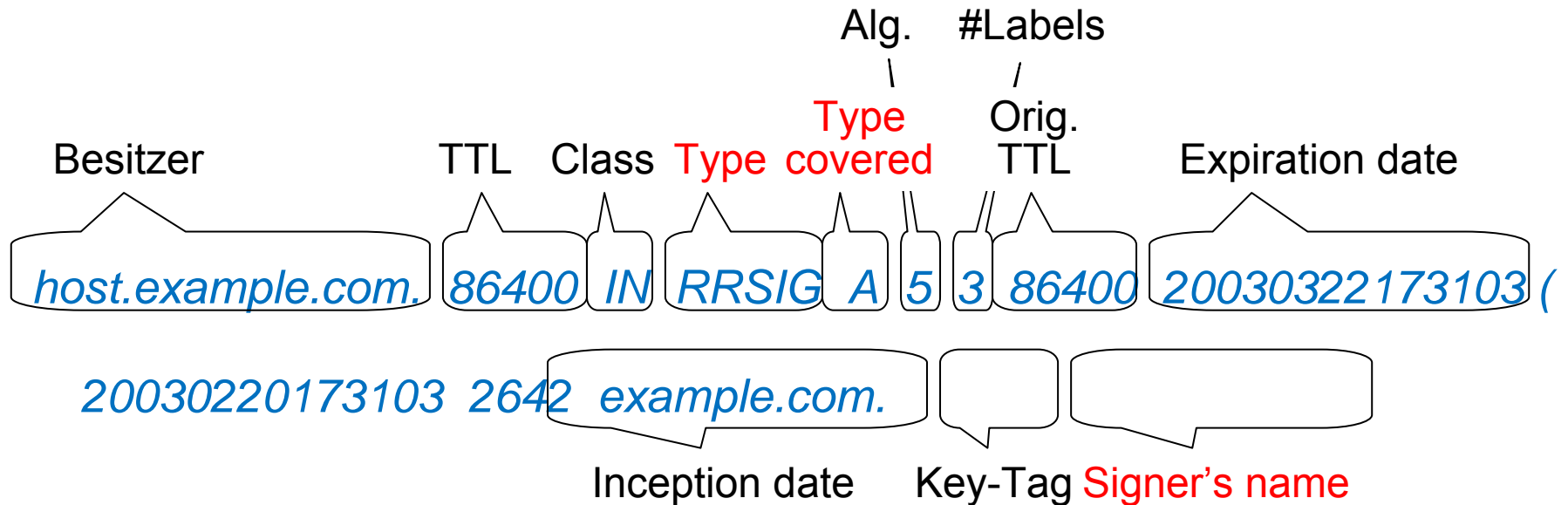
- Beispiel aus RFC 4034





# RR Signature (RRSIG)

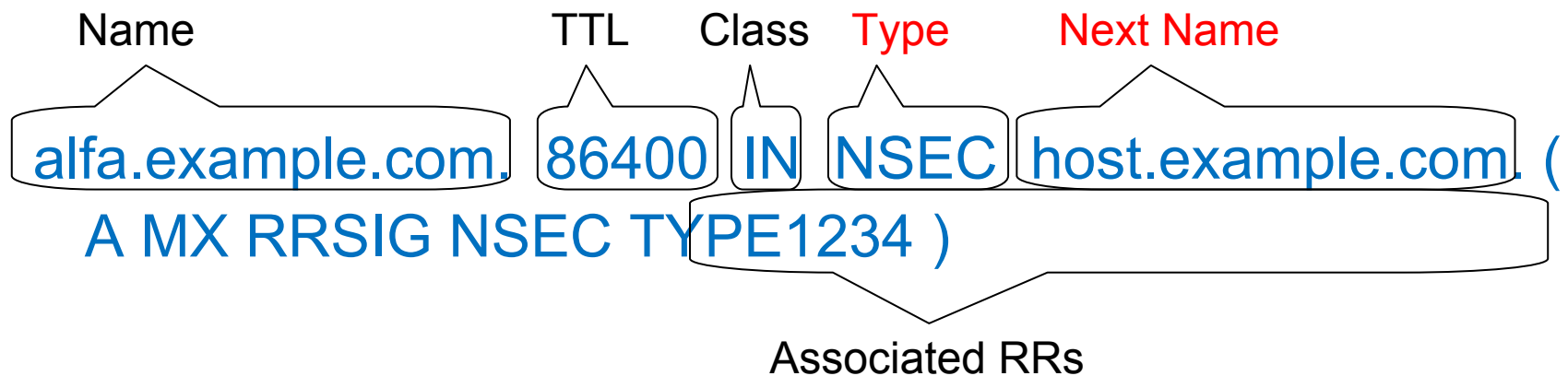
- Beispiel aus RFC 4034



oJB1W6WNGv+ldvQ3WDG0MQkg5IEhjRip8WTr  
 PYGv07h108dUKGMeDPKijVCHX3DDKdfb+v6o  
 B9wfu3T4XUAFI/M0zmO/zz8bW0Rznl8O3t  
 GNazPwQRRN20XPXV6nwwfoXmJQbsLNrLfkG  
 J5D6fwFm8nN+6pBzeDQfsS3Ap3o= )

**Signature**  
 Base64

- Beispiel aus RFC 4034



- Authentizität bestätigt durch RRSIG

- DNS Daten sind als öffentlich anzusehen
  - Gefahr: Information für neue Angriffe
- *“It's the difference between letting random folks call your company's switchboard and ask for John Q. Cubicle's phone number [versus] sending them a copy of your corporate phone directory”*
- DNSSec mit NSEC kann möglicherweise nicht legal in allen Ländern eingesetzt werden

- Zusicherung, dass ein Name nicht existiert ohne Namen der Zone offenzulegen
  - NSEC3 RR enthalten Hashwerte an Stelle von Name
- Anfrage vergleicht Hash des angefragten Namens mit den Hashwerten der Antwort
  - Hash ungleich der beiden Hashwerte → Name existiert nicht in der Zone





- Größere DNS-Pakete
  - Firewalls filtern DNS-Pakete größer als 512Byte
- Höhere Last auf Nameservern
  - Krypto-Operationen führen zu langsamern Auflösung
  - DDoS-Angriffe
- Anwendungsverhalten bei DNSSec Problem unklar
  - Definition von Fehlerverhalten
- Zeitsynchronisation der DNS-Server
- Erfahrung
  - Konfiguration
  - Betrieb

- [2.1] “**Off-the-record communication, or, why not to use PGP**”, Nikita Borisov, Ian Goldberg, Eric Brewer; Workshop On Privacy In The Electronic Society, Pages: 77 – 84, 2004; <http://doi.acm.org/10.1145/1029179.1029200>