

Domain Name System

Vortrag von Ingo Blechschmidt

am 5. Januar 2005

Gliederung

- Geschichte
- Design
- Lookup- und Record-Typen
- Zonentransfer
- Cache Poisoning
- Cache Snooping
- Speicherung beliebiger Daten im DNS

Geschichte

- Problem:
Zuordnung IPs $\leftrightarrow$ symbolische Namen
- Lösung:
Speicherung aller Zuordnungen in
hosts-Dateien:

```
$ cat /etc/hosts
216.239.59.104  google.de
213.239.211.178 pro-linux.de
...
```

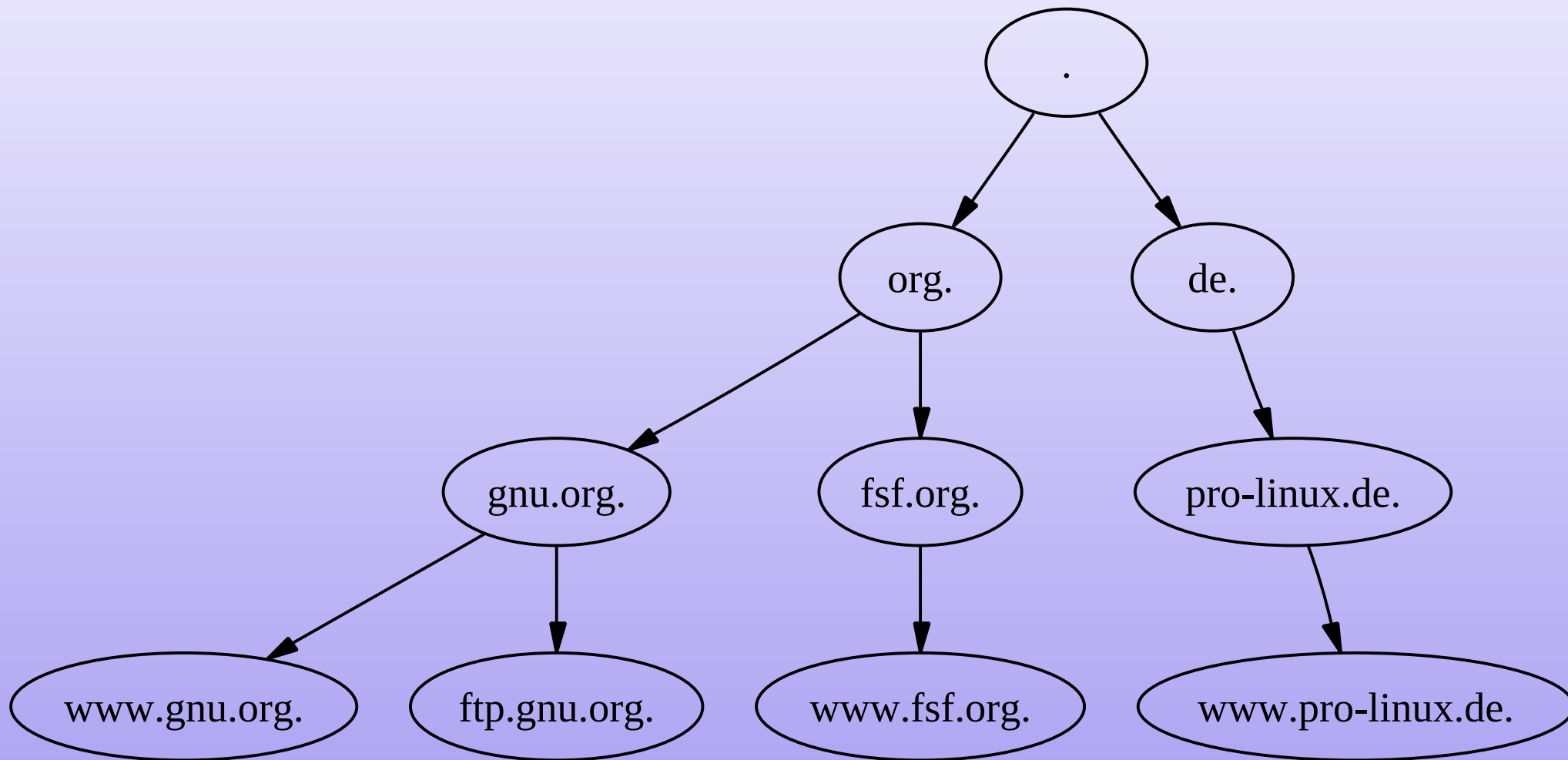
Geschichte

- Problem:
Bei jeder Aktualisierung Übermittlung an **jeden** Teilnehmer erforderlich
- Lösung:
Verwendung eines zentralen
Hostname-Servers (RFC 811, 1982):
`$ telnet zentraler-server 101`
`HNAME google.de`
`HOST : 216.239.59.104 : google.de`
`: CLUSTER : LINUX : TCP :`

Geschichte

- Problem:
Ausfall des zentralen Servers $\Rightarrow$
Zusammenbruch der Namensauflösung
- Endgültige Lösung:
Verwendung des Domain Name Systems

Design



Design

- Ausfall von . $\Rightarrow$
Zusammenbruch der Namensauflösung
Daher: Verteilung von . auf 13 Server
- Schlimmer noch:
Eigentümer von .: USA

Design

- Ausfall von . $\Rightarrow$
Zusammenbruch der Namensauflösung
Daher: Verteilung von . auf 13 Server
- Schlimmer noch:
Eigentümer von .: USA $\Rightarrow$
Zum Glück: Verwendung alternativer
Nameserver für . ,
z.B. Open Root Name Server Network

Lookup-Typen

- Iterativ („Durchhangeln“) :
 - „Hey . , was ist die IP von `www.pro-linux.de.`?“ –
„Ka, aber die IP des Nameservers von `de.` ist
`193.0.7.3`, frag' den.“

Lookup-Typen

- Iterativ („Durchhangeln“) :
 - „Hey . , was ist die IP von `www.pro-linux.de.`?“ – „Ka, aber die IP des Nameservers von `de.` ist `193.0.7.3`, frag' den.“
 - „Hey `de.` , was ist die IP von `www.pro-linux.de.`?“ – „Ka, aber die IP des Nameservers von `pro-linux.de.` ist `195.20.224.97`, frag' den.“

Lookup-Typen

- Iterativ („Durchhangeln“) :
 - „Hey . , was ist die IP von `www.pro-linux.de.`?“ – „Ka, aber die IP des Nameservers von `de.` ist `193.0.7.3`, frag' den.“
 - „Hey `de.` , was ist die IP von `www.pro-linux.de.`?“ – „Ka, aber die IP des Nameservers von `pro-linux.de.` ist `195.20.224.97`, frag' den.“
 - „Hey `pro-linux.de.` , was ist die IP von `www.pro-linux.de.`?“ – „Klar, die kenne ich, ich bin autoritativ für `pro-linux.de.` , die IP ist `213.239.211.178`.“

Lookup-Typen

- Iterativ („Durchhangeln“)
- Rekursiv:
Durchführung eines iterativen Lookups durch den angefragten rekursiven Nameserver und nur Übermittlung des End-Ergebnisses an den Client zurück –
„Hey, was ist die IP von `www.pro-linux.de`, und sag’ mir nicht, ich soll wo anders nachschauen!“

Record-Typen

- A: IP eines Hosts
\$ **dig** **www.pro-linux.de.** **A**
www.pro-linux.de. 8538 IN A 213.239.211.178
- NS: Nameserver einer Zone
- CNAME: „Symlink“ zu einem anderen Host
- MX: Mailserver einer Zone
- Zusätzlich:
ANY-Lookup $\Rightarrow$ Rückgabe aller Einträge

Record-Typen

- A: IP eines Hosts
- NS: Nameserver einer Zone
`$ dig pro-linux.de. NS`
`pro-linux.de. 8534 IN NS www.freesystems.de.`
`pro-linux.de. 8534 IN NS ns.schlund.de.`
- CNAME: „Symlink“ zu einem anderen Host
- MX: Mailserver einer Zone
- Zusätzlich:
ANY-Lookup $\Rightarrow$ Rückgabe aller Einträge

Record-Typen

- A: IP eines Hosts
- NS: Nameserver einer Zone
- CNAME: „Symlink“ zu einem anderen Host
- MX: Mailserver einer Zone

```
$ dig pro-linux.de. NS
```

```
pro-linux.de. 8431 IN MX 10 www.pro-linux.de.
```

- Zusätzlich:
ANY-Lookup $\Rightarrow$ Rückgabe aller Einträge

PTR-Record

- Problem:
Zuordnung IP $\rightarrow$ Name
- Lösung: Pointer-Records

PTR-Record

- Problem:
Zuordnung IP $\rightarrow$ Name
- Lösung: Pointer-Records
213 . 239 . 211 . 178

PTR-Record

- Problem:
Zuordnung IP $\rightarrow$ Name
- Lösung: Pointer-Records
178 . 211 . 239 . 213

PTR-Record

- Problem:
Zuordnung IP $\rightarrow$ Name
- Lösung: Pointer-Records
178.211.239.213.in-addr.arpa.

PTR-Record

- Problem:
Zuordnung IP $\rightarrow$ Name
- Lösung: Pointer-Records

```
$ dig 178.211.239.213.in-addr.arpa. PTR
178.211.239.213.in-addr.arpa. 86290 IN \
PTR www.pro-linux.de.
```

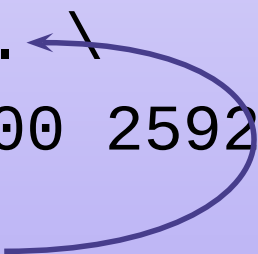
SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
- **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 2004112201 10800 1800 3600000 259200

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
- **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 2004112201 10800 1800 3600000 259200
- Primary der Zone,

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
 - **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 2004112201 10800 1800 3600000 259200
• Primary der Zone, Kontaktadresse,
- 

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
- **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 200411220110800 1800 3600000 259200
- Primary der Zone, Kontaktadresse,
 Seriennummer,

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
- **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 2004112201 10800 1800 3600000 259200
- Primary der Zone, Kontaktadresse,
Seriennummer, Slave-Refresh,

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone

- **\$ dig pro-linux.de. SOA**

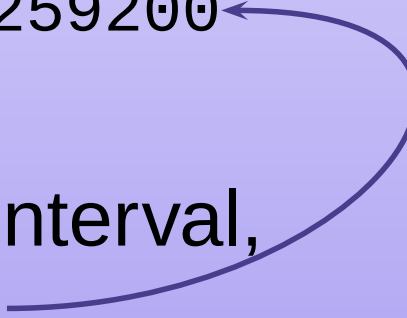
```
pro-linux.de. 82717 IN SOA \
    www.pro-linux.de. \
    postmaster.www.pro-linux.de. \
    2004112201 10800 1800 3600000 259200
```

- Primary der Zone, Kontaktadresse,
Seriennummer, Slave-Refresh, Retry-Interval,

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
- **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 2004112201 10800 1800 3600000 ← 259200
- Primary der Zone, Kontaktadresse,
Seriennummer, Slave-Refresh, Retry-Interval,
Expire-Timeout,

SOA-Record

- Start of Authority-Record:
Informationen über die gesamte Zone
 - **\$ dig pro-linux.de. SOA**
pro-linux.de. 82717 IN SOA \
 www.pro-linux.de. \
 postmaster.www.pro-linux.de. \
 2004112201 10800 1800 3600000 259200
 - Primary der Zone, Kontaktadresse,
Seriennummer, Slave-Refresh, Retry-Interval,
Expire-Timeout, TTL für NXDOMAINs
- 

Zonentransfer

- Mehrere Nameserver für eine Zone $\Rightarrow$ Redundanz
- Abgleich der Secondaries mit dem Primary durch Zonentransfer
- **\$ dig @zwinger.wikipedia.org \ wikipedia.org AXFR**
- U.U. Sicherheitsproblem
- Standardkonfiguration von BIND:
Zonentransfers von jedem Host aus erlaubt (!)

Caching Nameserver

- Erster Zugriff auf `pro-linux.de`.
- Rekursiver Lookup von `pro-linux.de`.
durch den Caching Nameserver des ISPs
- Caching der `pro-linux.de`-Records auf dem Nameserver
- Behalt der Records im Cache bis zum Ablauf der TTL

Cache Poisoning

- Annahme:

```
$ dig @ns1.google.com. google.com.
```

```
google.com.      300  IN  A  216.239.57.99
```

```
google.com.      300  IN  A  216.239.39.99
```

```
google.com.      300  IN  A  216.239.37.99
```

```
pro-linux.de.   300  IN  A  13.37.42.23
```

Cache Poisoning

- Annahme:

```
$ dig @ns1.google.com. google.com.
```

```
google.com.      300  IN  A  216.239.57.99
```

```
google.com.      300  IN  A  216.239.39.99
```

```
google.com.      300  IN  A  216.239.37.99
```

```
pro-linux.de.    300  IN  A  13.37.42.23
```

- \$ dig @cache google.com. >/dev/null

Cache Poisoning

- Annahme:

```
$ dig @ns1.google.com. google.com.
```

```
google.com.      300  IN  A  216.239.57.99
```

```
google.com.      300  IN  A  216.239.39.99
```

```
google.com.      300  IN  A  216.239.37.99
```

```
pro-linux.de.    300  IN  A  13.37.42.23
```

- \$ dig @cache google.com. >/dev/null

- \$ dig @cache pro-linux.de.

```
pro-linux.de.    300  IN  A  13.37.42.23
```

– Oops!

Cache Snooping

- Möglichkeit, nicht-rekursiven Lookup zu erzwingen $\Rightarrow$
- „Ist ein bestimmter Record bereits im Cache?“
`$ dig @ns www.pro-linux.de. +norecursive`
- Viele Fehlkonfigurationen $\Rightarrow$
Freier nicht-rekursiver Zugriff auf viele Caching Nameserver

Cache Snooping: Beispiele

- „Wurde in der letzten Zeit von einem bestimmten Provider aus auf eine Domain zugegriffen?“
\$ dig @ns-des-isps host +norecursive
- „Schreibt eine bestimmte Firma einer anderen Mails?“
- „Wo wohnt eine bestimmte Person?“

Cache Snooping: Beispiele

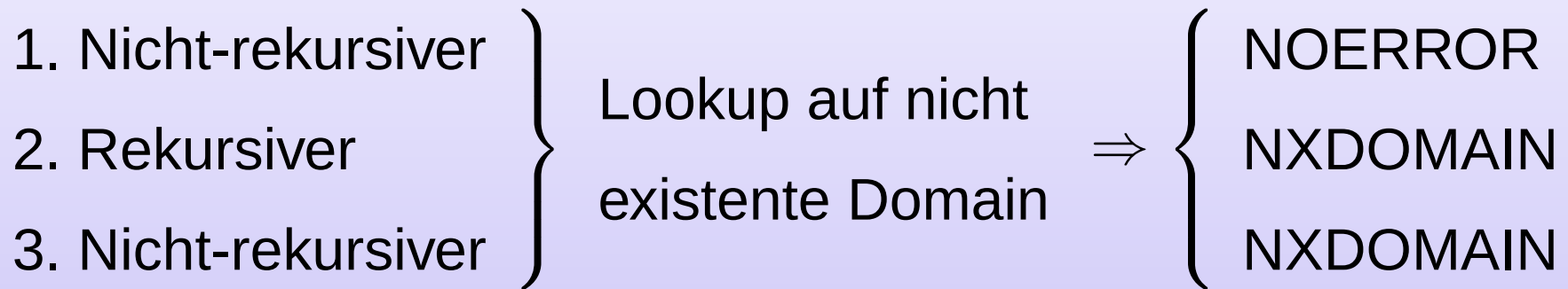
- „Wurde in der letzten Zeit von einem bestimmten Provider aus auf eine Domain zugegriffen?“
- „Schreibt eine bestimmte Firma einer anderen Mails?“

```
$ dig @ns-des-mailservers \
    mx.andere-firma.de. +norecursive
```
- „Wo wohnt eine bestimmte Person?“

Cache Snooping: Beispiele

- „Wo wohnt eine bestimmte Person?“
 - „Hey, geh’ doch mal auf `gibt-es-nicht.de`!“ –
 - „Kommt nur Fehler...“ –
 - „Ach, egal“
 - Nicht-rekursiver Lookup-Request von `gibt-es-nicht.de` an viele Nameserver $\Rightarrow$ NXDOMAIN-Antwort im Cache? $\Rightarrow$ ISP der Person gefunden

DNS als Speicher



DNS als Speicher

1. Nicht-rekursiver	} Lookup auf nicht existente Domain	$\Rightarrow$	NOERROR	$:= 0$
2. Rekursiver			NXDOMAIN	$:= 1$
3. Nicht-rekursiver			NXDOMAIN	$:= 1$

- Setzen eines Bits:

\$ dig @ns bit-addr +recursive

- Abfragen eines Bits:

\$ dig @ns bit-addr +norecursive $\Rightarrow$

NOERROR $\Rightarrow 0$

NXDOMAIN $\Rightarrow 1$

Sicherung von Servern

- Nicht-rekursiver Lookup $\Rightarrow$ Ignorieren des Cache
- Ablehnen von Zonentransfer-Requests von nicht-autorisierten Hosts
- Gute Cache-Algorithmen
- $\Rightarrow$ (In den meisten Fällen) nicht BIND, sondern djbdns oder dnscache

Siehe auch

- Hitchhiker's Guide to the Internet
<http://linide.sf.net/theguide2/>
- Open Root Server Network
<http://european.orcn.net/>
- Snooping the Cache for Fun and Profit
http://community.sidestep.pt/~luis/DNS-Cache-Snooping/DNS_Cache_Snooping_1.1.pdf

Siehe auch

- Proof-of-Concept Data-over-DNS
<http://m19s28.vlinux.de/iblech/dnsx.tar.bz2>
- D. J. Bernsteins djbdns
<http://cr.yp.to/djbdns.html>

Fragen?