Domain Name System – DNS

by

Dr. Günter Kolousek

DNS

- Zweck: Auflösung von Namen in IP-Adressen und umgekehrt
  - $\rightarrow$ Positionstransparenz, $\rightarrow$ Migrationstransparenz
- Struktur: verteiltes, hierarchisches System
- Namensraum
  - baumartig
    - Wurzel: .
  - in Zonen aufgeteilt
    - Zone: administrative Einheit
    - von DNS Servern verwaltet
    - Teilbäume werden oft an Subdomäne (mit eigenen Nameservern) *delegiert*

Komponenten

- Domainname (fully qualified domain name, FQDN)
  - besteht aus Folge von Labels (alphanumerisch und Bindestrich), getrennt durch .
- Nameserver (DNS Server)
- Resolver (DNS Client)
  - direkt in TCP/IP integriert
    - bekannte Programme: nslookup, host, dig, drill
  - Transportprotokoll defaultmäßig DNS
    - außer Zonentransfer oder Antwort zu groß $\rightarrow$ TCP
    - Port 53

Domainname

- DN
- Folge von Labels durch Punkte getrennt
  - Label: alphanumerisch (case-insensitive) und Bindestrich
    - max. 63 Bytes
    - TLD (top level domain)
  - abgeschlossen mit Punkt, kann weggelassen werden
  - max. 255 Bytes
- FQDN, z.B. `htlwrn.ac.at.`
- TLD…top level domain

Top level domain

- generische TLD
  - nicht gesponsert, z.B.: `.com`, `.info`, `.net`, `.org`, `.name`,...
    - haben ursprünglich eine Bedeutung gehabt und waren Einschränkungen unterworfen…
    - werden von ICANN (Internet Corporation for Assigned Names and Numbers) kontrolliert
  - gesponsert, z.B.: `.edu`, `.gov`, `.jobs`, `.tel`,...
- Länderspezifisch (gemäß ALPHA-2 von ISO-3166-1), z.B. `.at`, `.tv` (Tuvalu), `.st` (Sao Tome), `.io` (Britisches Territorium im Indischen Ozean),... und `.uk` sowie `.eu`
- neue TLDs, z.B. `.wien`, `.tirol`, `.bio`, `.suzuki`, `.gmail`, `.shop`, `.search`,...
- Spezial-TLDs, z.B. `.arpa`, `.localhost`, `.example`, `.test`,...
- Betrieb
  - jede TLD: Gruppe von Nameservern

Internationalisierter DN

- Internationalized Doman Name (IDN)
- Umlaute, chinesische Zeichen,…
- prinzipiell fast alle Unicode-Zeichen
  - jede Vergabestelle (network information center) schränkt ein
  - `nic.at` erlaubt: à á â ã ä å æ ç è é ê ë ì í î ï ð ñ ò ó ô õ ö ø œ š ù ú û ü ý ÿ ž þ
- zuerst meist Umwandlungen im Client wie z.B. Großbuchstaben in Kleinbuchstaben
- Umwandlung in ASCII mittels Punycode-Kodierung
  - Beispiel: `dömäin.example` → `xn--dmin-moa0i.example`

Nameserver und Zonen

- Nameserver
  - autoritative Namesever: sind für Zone verantwortlich
  - nicht-autoritative Nameserver: sind nicht für Zone verantwortlich
- Zonen
  - Zone ist administrative Einheit und für einen Teil des Baumes verantwortlich
  - jede Zone mind. 2 Nameserver

Typen von DNS-Server

- Master und Slave
  - je Zone mind. zwei autoritative Server
  - einer hat Rolle des Masters, die anderen sind Slaves
  - Zonendatei wird am Master geändert
  - Slave bekommt Kopie
  - Änderungen werden an Seriennummer in Zonendatei erkannt
- Caching-Server
  - keine autoritativen Antworten
- Forwarder

Namensauflösung

- Arten
  - rekursive Namensauflösung
    - kennt Namensserver die Antwort nicht, dann fragt dieser selbständig weiter
  - iterative Namensauflösung
    - kennt Namenserver die Antwort nicht, dann liefert dieser den nächsten Namensserver zurück
- welche Art verwendet wird, hängt von Flag in Anfrage ab
  - Rootserver akzeptieren nur iterative Anfragen
    - es gibt 13 Rootserver
- inverse Anfrage: 192.170.149.127 $\rightarrow$ 127.149.170.192.in-addr.arpa

Beispiel: Namensauflösung (Typ A)

```
$ drill www.htlwrn.ac.at
...
;; QUESTION SECTION:
;; www.htlwrn.ac.at.    IN      A

;; ANSWER SECTION:
www.htlwrn.ac.at.       16714   IN      A       195.202.147.97
...
```

Zonendatei und RR

- Zonendatei
  - (ursprünglich) Teil der Konfiguration von BIND
  - besteht aus
    - Liste von Resource Records (RR)
  - beschreibt Zone
- Resource Record
  - `<nam>` Domänenname des Objektes
  - `<ttl>` TTL (optional) $\rightarrow$ caching server!
  - `<class>` Protokollgruppe (optional), de facto nur `IN`
  - `<type>` Typ des RR
  - `<rdlength>` Länge der Daten (optional)
  - manche Typen erwarten sich hier weitere Felder (wie z.B. bei `MX`)
  - `<rdata>` Daten des RR

Typen von RR

- A … Address Record
- AAAA … für IPv6
- CNAME … Canonical Name, legt anderen Namen fest
  - Alias für anderen DN, z.B. `www.htlwrn.ac.at` → `htlwrn.ac.at`
- MX … Mail eXchange
- NS … Nameserver: Delegiert Subdomain zu Nameserver
  - z.B.: `htlwrn.ac.at.` → `venus.htlwrn.ac.at.`
- PTR … Pointer Record, d.h. für inverse Auflösung
- SOA … Start Of Authority, d.h. Informationen über Zone
- SRV … Service Locator, wird bei allgemeinen Diensten verwendet
  - z.B. wird von SPF (Sender Policy Framework, Spamabwehr)
- TXT … eigentlich für Menschen, heute für verschiedenste Dienste

Beispiel: Typ MX

```
$ drill htlwrn.ac.at MX
...
;; htlwrn.ac.at.          IN      MX

;; ANSWER SECTION:
htlwrn.ac.at.   85682   IN      MX      5 avispa2.htlwrn.ac.at.
htlwrn.ac.at.   85682   IN      MX      120 mail2.htlwrn.ac.at.
htlwrn.ac.at.   85682   IN      MX      100 mail.htlwrn.ac.at.
...
```

Beispiel: inverse Anfrage

```
$ drill -x 195.202.147.72
...
;; QUESTION SECTION:
;; 72.147.202.195.in-addr.arpa. IN      PTR

;; ANSWER SECTION:
72.147.202.195.in-addr.arpa.    38400   IN      PTR     mail2.htlwrn.a
...
```

Beispiel: beliebiger Typ

```
$ drill htlwrn.ac.at any
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 3239
;; flags: qr rd ra ; QUERY: 1, ANSWER: 8, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;; htlwrn.ac.at.        IN      ANY

;; ANSWER SECTION:
htlwrn.ac.at.   86392   IN      SRV     10 2 5061 _sipfederationtls._tcp.student.htlwrrn.ac.a
htlwrn.ac.at.   86392   IN      TXT     ”v=spf1 mx mx:avispa.htlwrn.ac.at mx:avispa1.htlwrn.a
htlwrn.ac.at.   86392   IN      MX      5 avispa2.htlwrn.ac.at.
htlwrn.ac.at.   86392   IN      MX      100 mail.htlwrn.ac.at.
htlwrn.ac.at.   86392   IN      MX      120 mail2.htlwrn.ac.at.
htlwrn.ac.at.   86392   IN      SOA     venus.htlwrn.ac.at. root.venus.htlwrn.ac.at. 2005080
htlwrn.ac.at.   8637    IN      NS      jupiter.htlwrn.ac.at.
htlwrn.ac.at.   8637    IN      NS      venus.htlwrn.ac.at.

;; AUTHORITY SECTION:

;; ADDITIONAL SECTION:

;; Query time: 25 msec
;; SERVER: 192.168.8.1
;; WHEN: Sun Sep  2 12:02:52 2018
;; MSG SIZE  rcvd: 470
```

Dynamisches DNS (DDNS)

- Zweck: dynamische Aktualisierung von DNS Einträgen
  - Szenario 1: Server werden nicht mit statischen IP Adressen versorgt, sondern mittels DHCP werden IP Adressen dynamisch vergeben ($\rightarrow$ Felixibilität).
  - Szenario 2: Rechner im Heimnetzwerk bekommen vom Provider dynamisch sich ändernde IP Adressen zugeteilt. Ein Host soll im Internet als Server dienen (z.B. Anbieter DynDNS)
- D.h. es sind Änderungen regelmäßig und automatisch im DNS vorzunehmen.
- 2 Möglichkeiten der Realisierung
  - DDNS über RFC 2136 (DNS Update): Protokoll hauptsächlich zwischen DHCP Server und DNS Server. Sicherheits-relevante Updates in RFC 2137 und RFC 3007.
  - DDNS über HTTP: Änderungen werden per HTTPS aktiv dem DDNS Anbieter bekanntgegeben. Dazu ist eine Client-SW am Host zu installieren.

Sicherheit

- DNSSEC
  - Sicherstellung der Integrität mittels asymmetrischen Verfahren
- DoT
  - DNS over TLS
  - z.B. Coudflare (1.1.1.1) oder Google (8.8.8.8)
  - aber auch: Digitalcourage (46.182.19.48)
- DoH
  - DNS over HTTPS
  - z.B. Coudflare (1.1.1.1) oder Google (8.8.8.8)
  - aber auch: Digitale Gesellschaft Schweiz (185.95.218.42)