10.3 DNS und Namensauflösung

DNS gehört zur OSI-Schicht 7.

DNS steht für:

```
Domain Name System
```

Die Hauptaufgabe von DNS ist:

```
Namen in IP-Adressen auflösen.
```

Beispiel:

```
www.example.com
→ 93.184.216.34
```

Menschen merken sich Namen leichter als IP-Adressen.

Computer benötigen aber IP-Adressen, um über das Netzwerk zu kommunizieren.

Merksatz:

```
DNS = Name zu IP-Adresse.
```

Warum braucht man DNS?

Ohne DNS müsste man sich IP-Adressen merken.

Beispiel ohne DNS:

```
93.184.216.34
```

Beispiel mit DNS:

```
www.example.com
```

DNS macht Netzwerke benutzerfreundlicher und flexibler.

Wenn sich die IP-Adresse eines Servers ändert, kann der DNS-Eintrag angepasst werden, ohne dass sich der Name ändern muss.

Merksatz:

```
DNS trennt Namen von IP-Adressen.
```

DNS im OSI-Modell

DNS ist ein Anwendungsschicht-Protokoll.

Es nutzt Transportprotokolle aus Schicht 4.

Typisch:

```
UDP 53
TCP 53
```

Wichtig:

```
DNS-Port 53 gehört zu Schicht 4.
DNS-Protokoll gehört zu Schicht 7.
```

Merksatz:

```
DNS nutzt Ports,
ist aber ein Dienst der Anwendungsschicht.
```

DNS und IP-Kommunikation

DNS ist nicht nötig, wenn direkt mit einer IP-Adresse kommuniziert wird.

Beispiel:

```
ping 8.8.8.8
```

Hier wird kein DNS-Name benötigt.

DNS wird benötigt bei:

```
ping www.example.com
```

Denn der Name muss zuerst in eine IP-Adresse aufgelöst werden.

Merksatz:

```
IP-Adresse direkt = kein DNS nötig.
Name verwenden = DNS nötig.
```

Typischer Ablauf einer Namensauflösung

Vereinfacht:

```
1. Benutzer gibt Namen ein.
2. Client prüft lokalen Cache.
3. Client fragt DNS-Resolver.
4. Resolver sucht die passende Antwort.
5. Resolver liefert IP-Adresse zurück.
6. Client verbindet sich zur IP-Adresse.
```

Beispiel:

```
Browser:
www.example.com öffnen

DNS:
www.example.com = 93.184.216.34

Browser:
Verbindung zu 93.184.216.34 aufbauen
```

Merksatz:

```
Erst Name auflösen,
dann Verbindung zur IP-Adresse aufbauen.
```

DNS-Client

Der DNS-Client ist das System, das eine DNS-Anfrage stellt.

Beispiele:

- PC
- Smartphone
- Server
- Browser
- Anwendung
- Betriebssystem

Der Client fragt normalerweise nicht direkt alle DNS-Server im Internet, sondern nutzt einen eingetragenen DNS-Resolver.

Merksatz:

DNS-Client stellt die Anfrage.

DNS-Resolver

Ein DNS-Resolver beantwortet DNS-Anfragen für Clients.

Er kann:

- Antworten aus Cache liefern
- andere DNS-Server fragen
- rekursive Auflösung durchführen
- lokale DNS-Zonen kennen

Beispiele für Resolver:

- Router im Heimnetz
- interner DNS-Server
- Provider-DNS
- öffentlicher DNS-Dienst

Merksatz:

Resolver sucht die Antwort für den Client.

Rekursive DNS-Abfrage

Bei einer rekursiven Abfrage erwartet der Client vom Resolver eine fertige Antwort.

Der Client fragt:

```
Welche IP hat www.example.com?
```

Der Resolver kümmert sich darum, die Antwort zu finden.

Der Client erhält am Ende:

```
IP-Adresse
oder
Fehler
```

Merksatz:

```
Rekursiv = Resolver erledigt die Suche für den Client.
```

Iterative DNS-Abfrage

Bei einer iterativen Abfrage liefert ein DNS-Server nicht unbedingt die endgültige Antwort.

Er kann stattdessen sagen:

```
Frag den nächsten zuständigen Server.
```

Das passiert typischerweise zwischen DNS-Servern bei der Auflösung.

Merksatz:

```
Iterativ = Server verweist auf nächsten zuständigen Server.
```

DNS-Hierarchie

DNS ist hierarchisch aufgebaut.

Vereinfacht:

```
Root
Top-Level-Domain
Domain
Subdomain
Hostname
```

Beispiel:

```
www.example.com
```

Aufteilung:

Teil	Bedeutung
.	Root
com	Top-Level-Domain
example	Domain
www	Hostname oder Subdomain

Merksatz:

```
DNS ist wie ein Baum aufgebaut.
```

Root-DNS-Server

Root-DNS-Server stehen ganz oben in der DNS-Hierarchie.

Sie kennen nicht jede einzelne Domain, aber sie wissen, welche Nameserver für Top-Level-Domains zuständig sind.

Beispiel:

```
Für .de frage diese Nameserver.
Für .com frage diese Nameserver.
```

Merksatz:

```
Root-Server verweisen auf TLD-Nameserver.
```

Top-Level-Domain

Eine Top-Level-Domain ist der rechte Teil eines Domainnamens.

Beispiele:

```
.de
.com
.org
.net
.edu
```

Bei:

```
www.example.com
```

ist:

```
.com
```

die Top-Level-Domain.

Merksatz:

```
TLD = oberster Domainbereich wie .de oder .com.
```

Authoritative DNS-Server

Ein authoritative DNS-Server ist für eine DNS-Zone zuständig.

Er enthält verbindliche Informationen für eine Domain.

Beispiel:

```
Der authoritative DNS-Server für example.com kennt Einträge wie:

www.example.com
mail.example.com
api.example.com
```

Merksatz:

```
Authoritative DNS-Server liefern verbindliche Antworten für eine Zone.
```

DNS-Zone

Eine DNS-Zone ist ein verwalteter Bereich im DNS.

Beispiel:

```
example.com
```

In dieser Zone können Einträge liegen wie:

```
www.example.com
mail.example.com
api.example.com
```

Eine Zone wird von zuständigen DNS-Servern verwaltet.

Merksatz:

```
DNS-Zone = verwalteter Namensbereich.
```

Forward Lookup

Forward Lookup bedeutet:

```
Name zu IP-Adresse
```

Beispiel:

```
www.example.com
→ 93.184.216.34
```

Das ist die häufigste DNS-Nutzung.

Merksatz:

```
Forward Lookup = Name zu IP.
```

Reverse Lookup

Reverse Lookup bedeutet:

```
IP-Adresse zu Name
```

Beispiel:

```
93.184.216.34
→ www.example.com
```

Dafür werden spezielle DNS-Zonen verwendet.

Bei IPv4:

```
in-addr.arpa
```

Bei IPv6:

```
ip6.arpa
```

Merksatz:

```
Reverse Lookup = IP zu Name.
```

DNS-Records

DNS speichert Informationen in Records.

Ein Record ist ein DNS-Eintrag.

Wichtige Record-Typen:

Record	Bedeutung
A	Name zu IPv4-Adresse

Record	Bedeutung
AAAA	Name zu IPv6-Adresse
CNAME	Alias auf anderen Namen
MX	Mailserver einer Domain
NS	zuständiger Nameserver
TXT	Textinformationen
PTR	Reverse-DNS-Eintrag
SRV	Dienststandort
SOA	Start of Authority

Merksatz:

```
DNS-Records speichern verschiedene Namensinformationen.
```

A-Record

Ein A-Record ordnet einen Namen einer IPv4-Adresse zu.

Beispiel:

```
www.example.com
→ 93.184.216.34
```

A steht für:

```
Address
```

Merksatz:

```
A-Record = Name zu IPv4.
```

AAAA-Record

Ein AAAA-Record ordnet einen Namen einer IPv6-Adresse zu.

Beispiel:

```
www.example.com
→ 2001:db8::10
```

Warum AAAA?

```
IPv6-Adressen sind 128 Bit groß.
IPv4-Adressen sind 32 Bit groß.
128 Bit sind viermal so groß wie 32 Bit.
```

Merksatz:

```
AAAA-Record = Name zu IPv6.
```

CNAME-Record

Ein CNAME ist ein Alias.

CNAME steht für:

```
Canonical Name
```

Beispiel:

```
shop.example.com
→ webserver.example.com
```

Der Name shop.example.com zeigt also auf einen anderen Namen.

Wichtig:

```
Ein CNAME zeigt auf einen Namen,
nicht direkt auf eine IP-Adresse.
```

Merksatz:

```
CNAME = Alias auf anderen DNS-Namen.
```

MX-Record

MX steht für:

```
Mail Exchanger
```

Ein MX-Record gibt an, welche Mailserver für eine Domain zuständig sind.

Beispiel:

```
example.com
MX 10 mail.example.com
```

Die Zahl ist die Priorität.

Kleinere Zahl bedeutet:

```
höhere Priorität
```

Merksatz:

```
MX = zuständiger Mailserver einer Domain.
```

NS-Record

NS steht für:

```
Name Server
```

Ein NS-Record gibt an, welche Nameserver für eine Zone zuständig sind.

Beispiel:

```
example.com
NS ns1.example.com
example.com
NS ns2.example.com
```

Merksatz:

```
NS = zuständiger Nameserver.
```

TXT-Record

TXT-Records speichern Textinformationen.

Typische Nutzung:

```
- SPF
- DKIM
- DMARC
- Domain-Verifizierung
- Sicherheitsinformationen
- Dienstnachweise
```

Beispiel:

```
v=spf1 include:example.net -all
```

Merksatz:

```
TXT-Record = Textinformationen im DNS.
```

PTR-Record

PTR steht für:

```
Pointer
```

Ein PTR-Record wird für Reverse Lookup verwendet.

Er ordnet eine IP-Adresse einem Namen zu.

Beispiel:

```
93.184.216.34
→ server.example.com
```

PTR-Records sind besonders bei Mailservern relevant.

Merksatz:

```
PTR = Reverse-DNS-Eintrag.
```

SRV-Record

SRV steht für:

```
Service
```

Ein SRV-Record beschreibt, wo ein bestimmter Dienst erreichbar ist.

Er kann enthalten:

```
- Dienst
- Protokoll
- Zielhost
- Port
- Priorität
- Gewichtung
```

Typische Nutzung:

```
- Verzeichnisdienste
- VoIP
- Microsoft-Dienste
- XMPP
```

Merksatz:

```
SRV zeigt,
wo ein Dienst erreichbar ist.
```

SOA-Record

SOA steht für:

```
Start of Authority
```

Der SOA-Record enthält Verwaltungsinformationen einer DNS-Zone.

Dazu gehören zum Beispiel:

```
- primärer Nameserver
- verantwortliche Kontaktadresse
- Seriennummer
- Aktualisierungszeiten
- Ablaufzeiten
```

Merksatz:

```
SOA enthält Verwaltungsdaten einer DNS-Zone.
```

TTL

TTL steht bei DNS für:

```
Time To Live
```

Die TTL gibt an, wie lange ein DNS-Eintrag zwischengespeichert werden darf.

Beispiel:

```
TTL 3600
```

Bedeutung:

```
Eintrag darf 3600 Sekunden,
also eine Stunde,
gecacht werden.
```

Merksatz:

DNS-TTL bestimmt,
wie lange eine Antwort im Cache bleiben darf.

DNS-Cache

Ein DNS-Cache speichert Antworten vorübergehend.

Caches gibt es zum Beispiel bei:

- Betriebssystem
- Browser
- DNS-Resolver
- Router
- Provider
- Anwendungen

Vorteil:

schnellere Antworten
weniger Last auf DNS-Servern

Nachteil:

Änderungen werden nicht sofort überall sichtbar.

Merksatz:

DNS-Cache macht DNS schneller,
kann Änderungen aber verzögern.

Warum DNS-Änderungen dauern können

DNS-Änderungen sind nicht immer sofort überall sichtbar.

Grund:

alte Antworten können noch in Caches liegen.

Die Dauer hängt stark von der TTL ab.

Beispiel:

```
TTL 86400
```

bedeutet:

```
Antwort darf bis zu 24 Stunden gecacht werden.
```

Merksatz:

```
DNS-Änderungen brauchen wegen Caching Zeit.
```

DNS und DHCP

Clients erhalten DNS-Server häufig per DHCP.

DHCP kann dem Client mitteilen:

```
- IP-Adresse
- Subnetzmaske
- Gateway
- DNS-Server
- Domain-Suffix
```

Wenn der falsche DNS-Server per DHCP verteilt wird, funktioniert Namensauflösung möglicherweise falsch.

Merksatz:

```
DHCP verteilt oft die DNS-Serveradresse.
```

DNS-Suffix

Ein DNS-Suffix kann bei unvollständigen Namen ergänzt werden.

Beispiel:

```
eingegeben:
server01
```

DNS-Suffix:

```
firma.local
```

Abfrage wird zu:

```
server01.firma.local
```

Typisch in Unternehmensnetzen.

Merksatz:

```
DNS-Suffix ergänzt kurze Namen.
```

FQDN

FQDN steht für:

```
Fully Qualified Domain Name
```

Ein FQDN ist ein vollständiger DNS-Name.

Beispiel:

```
server01.firma.local
```

Er beschreibt den Namen eindeutig innerhalb der DNS-Hierarchie.

Merksatz:

```
FQDN = vollständiger DNS-Name.
```

Hostname und Domainname

Hostname:

```
Name eines Geräts oder Dienstes
```

Domainname:

```
Namensbereich
```

Beispiel:

```
server01.firma.local
```

Dabei ist:

```
server01 = Hostname
firma.local = Domain
```

Merksatz:

```
Hostname ist der einzelne Name,
Domain ist der Namensbereich.
```

Split DNS

Split DNS bedeutet:

```
Ein Name kann intern und extern unterschiedlich aufgelöst werden.
```

Beispiel:

```
wiki.firma.de
```

intern:

```
192.168.10.50
```

extern:

```
93.184.100.10
```

Das ist häufig sinnvoll, wenn interne Clients interne Adressen verwenden sollen.

Merksatz:

```
Split DNS liefert je nach Standort unterschiedliche Antworten.
```

Interne und externe DNS-Zonen

Unternehmen nutzen oft interne und externe DNS-Zonen.

Extern sichtbar:

```
www.firma.de
mail.firma.de
```

Intern zusätzlich:

```
server01.firma.local
db01.intern.firma.de
printer01.firma.local
```

Wichtig:

```
Interne Namen sollten nicht unkontrolliert öffentlich sichtbar sein.
```

Merksatz:

```
Interne DNS-Zonen dienen interner Namensauflösung.
```

DNS und Active Directory

In Windows-Domänen ist DNS besonders wichtig.

Active Directory nutzt DNS für:

- Domänencontroller finden
- Dienste finden
- LDAP finden
- Kerberos finden
- Anmeldung unterstützen

Dafür werden unter anderem SRV-Records verwendet.

Merksatz:

> Active Directory braucht funktionierendes DNS.

DNS und E-Mail

DNS ist für E-Mail sehr wichtig.

Wichtige Records:

Record	Zweck
MX	zuständiger Mailserver
A / AAAA	IP-Adresse des Mailservers
PTR	Reverse-DNS für Mailserver
TXT	SPF, DKIM, DMARC

Ohne korrekte DNS-Einträge kann E-Mail-Zustellung scheitern.

Merksatz:

> Mailzustellung hängt stark von DNS ab.

SPF

SPF steht für:

> Sender Policy Framework

SPF wird über TXT-Records im DNS veröffentlicht.

SPF legt fest, welche Server für eine Domain E-Mails senden dürfen.

Merksatz:

```
SPF sagt,
welche Server für eine Domain senden dürfen.
```

DKIM

DKIM steht für:

```
DomainKeys Identified Mail
```

DKIM nutzt digitale Signaturen, um E-Mails einer Domain zuzuordnen und Manipulationen erkennbar zu machen.

Der öffentliche Schlüssel steht im DNS als TXT-Record.

Merksatz:

```
DKIM signiert E-Mails kryptografisch.
```

DMARC

DMARC steht für:

```
Domain-based Message Authentication, Reporting and Conformance
```

DMARC baut auf SPF und DKIM auf.

Es legt fest, wie Empfänger mit verdächtigen E-Mails umgehen sollen.

Beispiele:

```
- nur berichten
- in Quarantäne verschieben
- ablehnen
```

Merksatz:

```
DMARC sagt,
wie mit SPF-/DKIM-Fehlern umzugehen ist.
```

DNS und Webseiten

Für Webseiten sind häufig wichtig:

```
A-Record
AAAA-Record
CNAME
TXT für Domainprüfung
```

Beispiel:

```
www.firma.de
→ A-Record oder CNAME
```

Wenn DNS auf die falsche IP zeigt, landet der Browser auf dem falschen Server.

Merksatz:

```
Falsches DNS kann falsche Webseite oder falsches Zertifikat bedeuten.
```

DNS und Zertifikate

TLS-Zertifikate müssen zum aufgerufenen Namen passen.

Wenn DNS auf einen falschen Server zeigt, kann dieser ein falsches Zertifikat ausliefern.

Beispiel:

```
DNS für wiki.firma.de zeigt auf Server A.
Server A liefert Zertifikat für cloud.firma.de.
```

Folge:

```
Browser zeigt Zertifikatswarnung.
```

Merksatz:

```
DNS und Zertifikate müssen zusammenpassen.
```

DNS und Reverse Proxy

Bei einem Reverse Proxy zeigen DNS-Einträge häufig auf den Proxy.

Beispiel:

```
wiki.firma.de
cloud.firma.de
git.firma.de
```

zeigen alle auf:

```
Reverse Proxy
```

Der Proxy entscheidet anhand des Hostnamens, an welches interne Backend weitergeleitet wird.

Merksatz:

```
DNS zeigt oft zum Proxy,
der Proxy leitet zum Backend.
```

DNS-Fehlersuche

Typische Fragen:

```
Welcher DNS-Server wird verwendet?
Wird der Name korrekt aufgelöst?
Gibt es A- oder AAAA-Records?
Gibt es einen falschen CNAME?
Ist der Cache veraltet?
Ist die TTL noch aktiv?
Wird intern anders aufgelöst als extern?
Stimmt der MX-Record?
Stimmt der PTR-Record?
```

```
Gibt es Tippfehler im Namen?
```

Merksatz:

```
DNS-Fehlersuche prüft Server,
Record,
Cache
und Sichtweise.
```

Fehlerbild: IP funktioniert, Name nicht

Beispiel:

```
ping 8.8.8.8 funktioniert.
ping www.example.com funktioniert nicht.
```

Dann ist IP-Kommunikation grundsätzlich möglich.

Mögliche Ursachen:

```
- DNS-Server falsch
- DNS-Server nicht erreichbar
- DNS-Dienst gestört
- Firewall blockiert DNS
- falscher Domainname
- DNS-Cache fehlerhaft
```

Merksatz:

```
IP geht,
Name nicht:
DNS prüfen.
```

Fehlerbild: Name zeigt auf falsche IP

Mögliche Ursachen:

- falscher A-Record
- falscher AAAA-Record
- falscher CNAME
- alter Cache
- falsche DNS-Zone
- Split-DNS falsch
- falscher DNS-Server wird gefragt

Merksatz:

Falsche IP bei DNS:
Record und verwendeten DNS-Server prüfen.

Fehlerbild: Intern geht, extern nicht

Mögliche Ursachen:

- interne DNS-Zone anders als externe
- externer DNS-Eintrag fehlt
- Firewall blockiert extern
- NAT oder Portweiterleitung fehlt
- Zertifikat gilt nur intern
- öffentliche IP falsch
- Split DNS falsch geplant

Merksatz:

Intern und extern getrennt prüfen.

Fehlerbild: Extern geht, intern nicht

Mögliche Ursachen:

- interner DNS-Server liefert falsche Adresse
- Hairpin NAT fehlt
- Split DNS fehlt
- interne Firewall blockiert

- interner Proxy falsch
- interne Zone überschreibt externe Zone

Merksatz:

Extern erreichbar heißt nicht automatisch intern erreichbar.

Fehlerbild: DNS-Änderung greift nicht

Mögliche Ursachen:

- DNS-Cache noch aktiv
- TTL noch nicht abgelaufen
- falscher DNS-Server wird gefragt
- Änderung in falscher Zone gemacht
- autoritativer Nameserver nicht aktualisiert
- Browser oder Betriebssystem cachet noch

Merksatz:

DNS-Änderungen brauchen Zeit wegen TTL und Cache.

Fehlerbild: Webseite zeigt falsches Zertifikat

Mögliche Ursachen:

- DNS zeigt auf falsche IP
- Reverse Proxy liefert falsches Zertifikat
- Host-Header passt nicht
- falscher virtueller Host
- CNAME zeigt unerwartet weiter
- interne und externe DNS-Antwort unterscheiden sich

Merksatz:

Falsches Zertifikat kann DNS- oder Proxy-Ursache haben.

DNS-Tools

Typische Werkzeuge zur DNS-Prüfung:

Werkzeug	Zweck
nslookup	DNS-Abfragen durchführen
dig	detaillierte DNS-Abfragen
host	einfache DNS-Abfrage
ping	zeigt oft aufgelöste IP, prüft aber ICMP
ipconfig /displaydns	DNS-Cache unter Windows anzeigen
ipconfig /flushdns	DNS-Cache unter Windows leeren
resolvectl	DNS-Informationen unter vielen Linux-Systemen
scutil --dns	DNS-Informationen unter macOS

Merksatz:

```
DNS prüft man mit DNS-Werkzeugen,
nicht nur mit Ping.
```

Ping ist kein DNS-Test allein

Ping kann einen Namen auflösen, aber Ping prüft zusätzlich ICMP.

Beispiel:

```
ping www.example.com
```

Dabei passieren zwei Dinge:

```
1. DNS-Auflösung
2. ICMP-Test
```

Wenn Ping fehlschlägt, muss man unterscheiden:

```
DNS-Auflösung fehlgeschlagen?
Oder ICMP blockiert?
```

Merksatz:

```
Ping mit Name vermischt DNS und ICMP.
```

DNS und UDP/TCP 53

DNS nutzt häufig UDP 53.

DNS kann aber auch TCP 53 verwenden.

TCP 53 ist wichtig bei:

```
- großen Antworten
- Zonentransfers
- bestimmten DNSSEC-Fällen
```

Prüfungsfalle:

```
DNS ist nicht nur UDP.
```

Merksatz:

```
DNS nutzt UDP 53 häufig,
TCP 53 ebenfalls möglich.
```

Zonentransfer

Ein Zonentransfer überträgt DNS-Zonendaten zwischen DNS-Servern.

Typisch:

```
primärer DNS-Server
→ sekundärer DNS-Server
```

Zonentransfers sollten nicht öffentlich für jeden erlaubt sein.

Warum?

```
Sonst könnten interne DNS-Strukturen sichtbar werden.
```

Merksatz:

```
Zonentransfer nur für berechtigte DNS-Server erlauben.
```

DNSSEC kurz erklärt

DNSSEC steht für:

```
Domain Name System Security Extensions
```

DNSSEC schützt DNS-Antworten durch digitale Signaturen.

Ziel:

```
Manipulation von DNS-Antworten erkennbar machen.
```

Wichtig:

```
DNSSEC verschlüsselt DNS-Anfragen nicht automatisch.
Es schützt vor allem die Integrität und Authentizität der DNS-Daten.
```

Merksatz:

```
DNSSEC signiert DNS-Daten,
verschlüsselt sie aber nicht.
```

DNS over HTTPS und DNS over TLS

Es gibt auch verschlüsselte DNS-Varianten.

Beispiele:

```
DoH = DNS over HTTPS
DoT = DNS over TLS
```

Ziel:

DNS-Anfragen besser vor Mitlesen oder Manipulation auf dem Transportweg schützen.

Wichtig:

In Unternehmensnetzen kann das Auswirkungen auf Filterung und Monitoring haben.

Merksatz:

DoH und DoT verschlüsseln DNS-Transport.

Was DNS nicht macht

DNS macht nicht:

- Webseiten ausliefern
- TCP-Verbindungen aufbauen
- IP-Pakete routen
- Zertifikate ausstellen
- Dienste automatisch starten
- Firewall-Regeln setzen
- Benutzer authentifizieren

DNS liefert Namen und zugehörige Informationen.

Merksatz:

DNS sagt,
wohin ein Name zeigt,
nicht ob der Dienst funktioniert.

Einordnung in das OSI-Modell

Thema	Schicht
DNS-Protokoll	7
DNS-Record	7
A / AAAA / MX / TXT	7

Thema	Schicht
UDP 53 / TCP 53	4
IP-Adresse	3
Reverse Lookup	7
DNSSEC	7 mit Sicherheitsbezug
DoH / DoT	7 über TLS/HTTPS
DHCP verteilt DNS-Server	7 mit Schicht-3-Bezug

Merksatz:

> DNS gehört zu Schicht 7,
> nutzt aber Schicht 4 und Schicht 3 darunter.

Typische IHK-Fragen

In AP1 und AP2 kann zum Beispiel gefragt werden:

- Was macht DNS?
- Wofür steht DNS?
- Warum braucht man DNS?
- Was ist ein DNS-Resolver?
- Was ist ein authoritative DNS-Server?
- Was ist eine DNS-Zone?
- Was ist der Unterschied zwischen Forward und Reverse Lookup?
- Was ist ein A-Record?
- Was ist ein AAAA-Record?
- Was ist ein CNAME?
- Was ist ein MX-Record?
- Was ist ein TXT-Record?
- Was ist ein PTR-Record?
- Was bedeutet TTL?
- Warum dauern DNS-Änderungen manchmal?
- Warum funktioniert IP, aber Name nicht?
- Warum kann falsches DNS ein Zertifikatsproblem verursachen?
- Warum nutzt DNS UDP und TCP 53?

Typische Prüfungsfallen

DNS gehört zu Schicht 7.

DNS-Port 53 gehört zu Schicht 4.

DNS löst Namen in IP-Adressen auf.

A-Record = IPv4.

AAAA-Record = IPv6.

CNAME = Alias auf anderen Namen.

MX = Mailserver.

TXT = Textinformationen.

PTR = Reverse Lookup.

NS = zuständiger Nameserver.

TTL steuert DNS-Caching.

DNS-Änderungen können wegen Cache dauern.

IP geht,
Name nicht:
DNS prüfen.

Ping mit Name ist nicht nur DNS-Test,
sondern auch ICMP-Test.

DNS ist nicht nur UDP,
sondern kann auch TCP 53 nutzen.

DNSSEC verschlüsselt DNS nicht automatisch.

DNS sagt nicht,
ob der Dienst läuft.

Wichtige Begriffe kurz erklärt

Begriff	Kurze Erklärung
DNS	Domain Name System
Namensauflösung	Name wird zu IP-Adresse aufgelöst
Resolver	DNS-Server, der Antworten für Clients sucht
authoritative DNS	zuständiger DNS-Server einer Zone
DNS-Zone	verwalteter Namensbereich
Record	DNS-Eintrag
A	Name zu IPv4
AAAA	Name zu IPv6
CNAME	Alias auf anderen Namen
MX	Mailserver
NS	Nameserver
TXT	Textinformation
PTR	Reverse-DNS-Eintrag
SRV	Dienststandort
SOA	Verwaltungsdaten einer Zone
TTL	Cache-Lebensdauer
FQDN	vollständiger DNS-Name
Split DNS	unterschiedliche interne und externe Antworten
Reverse Lookup	IP-Adresse zu Name
DNSSEC	signierte DNS-Daten
DoH	DNS über HTTPS
DoT	DNS über TLS

IHK-sichere Kurzformulierung

DNS steht für Domain Name System und gehört zur Anwendungsschicht des OSI-Modells. DNS löst Namen in IP-Adressen auf, damit Anwendungen mit verständlichen Namen arbeiten können. Ein Client fragt einen DNS-Resolver, der die passende Antwort liefert oder weitere DNS-Server abfragt. DNS ist hierarchisch aufgebaut und arbeitet mit Records wie A für IPv4, AAAA für IPv6, CNAME für Aliase, MX für Mailserver, NS für Nameserver, TXT für Textinformationen und PTR für Reverse Lookup. DNS nutzt typischerweise UDP-Port 53, kann aber auch TCP-Port 53 verwenden. DNS-Caching wird durch die TTL gesteuert.

Merksätze

DNS = Domain Name System.

DNS gehört zu Schicht 7.

DNS löst Namen in IP-Adressen auf.

Name zu IP = Forward Lookup.

IP zu Name = Reverse Lookup.

Resolver sucht Antwort für Client.

Authoritative DNS liefert verbindliche Antwort.

DNS-Zone = verwalteter Namensbereich.

A = IPv4.

AAAA = IPv6.

CNAME = Alias.

MX = Mailserver.

NS = Nameserver.

TXT = Textinformation.

PTR = Reverse Lookup.

SRV = Dienststandort.

SOA = Zonendaten.

TTL = Cache-Zeit.

DNS-Cache beschleunigt,
verzögert aber Änderungen.

FQDN = vollständiger DNS-Name.

Split DNS = intern und extern unterschiedliche Antworten.

DNS-Port = UDP/TCP 53.

DNS ist nicht nur UDP.

IP geht,
Name nicht:
DNS prüfen.

Ping mit Name vermischt DNS und ICMP.

DNSSEC signiert,
verschlüsselt aber nicht automatisch.

DoH und DoT verschlüsseln DNS-Transport.

DNS sagt,
wohin ein Name zeigt,
nicht ob der Dienst funktioniert.