DHCP / DNS / FTP / TCP / UDP

DHCP

Dynamic Host Configuration Protocol – mit diesem Protokoll werden die IP-Adressen an Endgeräte (das könnte ein PC, Smartphone oder Drucker sein) innerhalb eines Netzwerks zugewiesen. Die Hosts werden dynamisch (automatisch) konfiguriert.

Das Ziel von diesem Protokoll ist es, dass jedem Host im Netzwerk eine IP-Adresse sowohl auch ein Gateway, Subnetzmaske und DNS automatisch zuzuordnen.

Port **67** - UDP (**Server**)
Port **68** - UDP (**Client**)

Layer 7 (Application Layer) - OSI Modell

Ablauf DHCP

1. DHCP**REQUEST**: Der Client fordert eine der angebotenen IP-Adressen, weitere Daten sowie Verlängerung der Lease-Zeit von einem der antwortenden DHCP-Server.
2. DHCP**OFFER**: Die DHCP-Server antworten mit entsprechenden Werten auf eine DHCPDISCOVER-Anfrage.
3. DHCP**ACK**: Bestätigung des DHCP-Servers zu einer DHCPREQUEST-Anforderung oder die Übermittlung von Konfigurationsparametern, die vorher durch DHCPINFORM vom Client angefordert wurden.
4. DHCP**NAK**: Ablehnung einer DHCPREQUEST-Anforderung durch den DHCP-Server.
5. DHCP**DECLINE**: Ablehnung durch den Client, da die IP-Adresse schon verwendet wird.
6. DHCP**RELEASE**: Der Client gibt die eigene Konfiguration frei, damit die Parameter wieder für andere Clients zur Verfügung stehen.
7. DHCP**INFORM**: Anfrage eines Clients nach weiteren Konfigurationsparametern, z. B. weil der Client eine statische IP-Adresse besitzt.

Wichtige Headereinträge DHCP

Client-/Transaction-ID	Zufallszahl, die der Client bestimmt und dem Server der Zuordnung hilft
Flags	Discover Offer Request Acknowledge
Server-Address	Hilft dem DHCP-Server zu erkennen, an wen der Request gerichtet ist
Client-Address	Die IP-Adresse, die dem Client dann zugeordnet wird
Client hardware address	MAC-Adresse welcher der Client Broadcastet

---

DNS

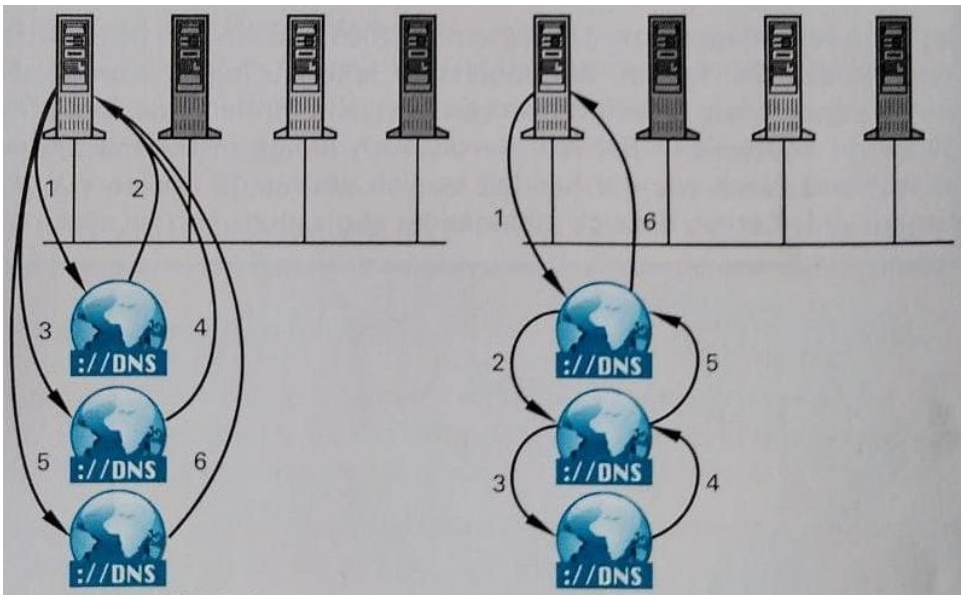
Domain Name System – funktioniert ähnlich wie ein Telefonbuch: Es verwaltet die Zuweisung zwischen Namen und Nummern. DNS-Server übersetzen Namensanforderungen in IP-Adressen und steuern dabei, welchen Server ein Endbenutzer erreicht, wenn er in seinen Webbrowser einen Domänen-Namen eingibt.

Port **53** - UDP

Layer 7 (Application Layer) - OSI Modell

Auf jedem Client ist ein Stück Software installiert, dass sich Resolver nennt. Sollte man eine IP-Adresse auf Basis einer URL benötigen, dann kümmert sich der Resolver um die Anfrage beim DNS-Server.

Eine Anfrage bei einem DNS-Server kann zwei unterschiedliche Antworten zur Folge haben.
Beim itterativen Verfahren (links im Bild) bekommt man entweder die gewünschte Antwort oder einen Verweis zu einem anderen DNS-Server. Dieser wird dann vom Resolver auf dem Client angefragt.
Beim rekursiven Verfahren (rechts im Bild) frägt der Resolver ebenfalls bei einem DNS-Server an, sollte der die Antwort nicht kennen, dann leitet dieser sie selbst an den nächsten DNS-Server weiter.

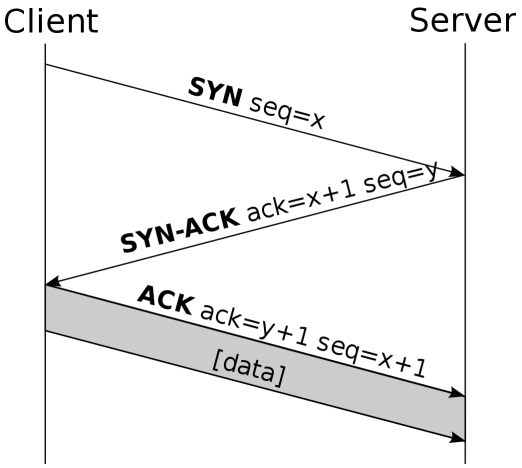


TCP

Transmission Control Protocol - ist ein verbindungsorientiertes und paketvermitteltes Transportprotokoll. Kommunikation über TCP lässt sich in drei Schritte einteilen, dem Verbindungsaufbau, der Datenübertragung und dem Verbindungsabbau.

Layer 7 (Application Layer) - OSI Modell

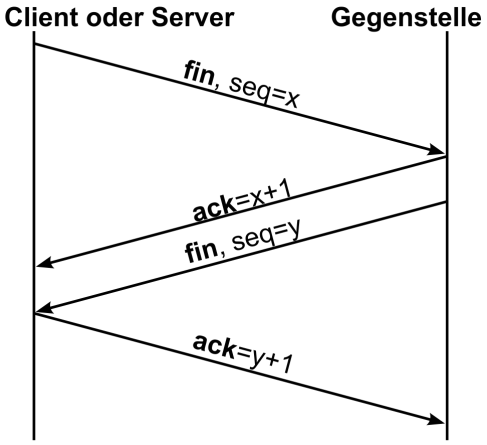
Verbindungsaufbau - Drei-Wege-Handshake
Client (Rechner der Verbindung aufbauen will) schickt ein **SYN** mit einer zufälligen 32-Bit langen Sequenznummer(seq). Ist der Server (Rechner zu dem die Verbindung aufgebaut wird) bereit die Verbindung zu zulassen, dann antwortet er mit einem **SYN-ACK**.Dabei wird die erhaltene seq um eins weitergezählt und als Acknowledgenummer(ack) zurück geschickt. Zusätzlich erstellt der Server selbst eine 32-Bit lange Sequenznummer und schickt diese mit.
Das **SYN-ACK** wird vom Client mit einem **ACK** bestätigt. Hier wird die seq des Servers als ack verwendet und um eins hochgezählt.



Datenübertragung
Hier werden die eigentlichen Informationen übertragen. Früher wurde jedes Paket das

ausgetauscht wurde von der Gegenseite mit Hilfe einer Sequenznummer bestätig. Das führt allerdings zu einer hohen Netzbelastung. Aus diesem Grund hat man die Möglichkeit geschaffen auch mehrere Pakete auf einmal schicken und bestätigen zu können. Dabei wird immer das letzte Paket, dass passt bestätigt. Damit ist dem Sender bekannt, was er nochmal nachschicken muss bzw. ab welchem Punkt er weitere Pakete schicken kann.

Verbindungsabbau - Vier-Wege-Handshake
Ähnlich wie beim Verbindungsaufbau, wird auch hier gegenseitig bestätig, dass man die Verbindung abbaut. Dabei schickt die eine Seite ein **FIN** mit zufälliger Sequenznummer(seq). Diese wird um eins hoch gezählt und als **ACK** zurückgeschickt. Zusätzlich wird ein zweites, eignes **FIN** Paket verschickt, welches ebenfalls mit einem **ACK** bestätigt wird. Erst ab diesem Moment ist die Verbindung geschlossen.



UDP

User Datagram Protocol - minimales und verbindungsloses Netzwerkprotokoll. Im Gegensatz zu TCP wird hier keine Verbindung aufgebaut, sondern die Daten einfach losgeschickt. Es gibt keine Bestätigung ob die Pakete angekommen sind oder nicht. Es kann also zu unbemerktem Datenverlust kommen. Jedoch gibt es eine Checksumme, die überprüft ob die Pakete fehlerfrei angekommen sind.
Streaming und Telefonie läuft über UDP, da der Verlust von einzelnen Paketen nicht wahrgenommen wird. Und sollte man doch etwas nicht verstanden haben kann man nachfragen.

Layer 7 (Application Layer) - OSI Modell

FTP

File Transfer Protocol - Protokoll wird genutzt um Daten von einem Client zum Server hochzuladen oder von einem Server zum Client herunterzuladen. Zusätzlich können über FTP auch Verzeichnisse angelegt und ausgelesen werden. Zudem können Verzeichnisse und Dateien umbenannt und gelöscht werden.

Aktives FTP - Der Client öffnet einen zufälligen Port und teilt diesen zusammen mit der eigenen IP-Adresse mit.

Passives FTP - Der Client bittet den Server eine Verbindung aufzubauen, daraufhin öffnet der Server einen Port und schickt diesen zusammen mit der Server-IP an den Client.

Der Verbindungsaufbau dient u.a. der Datenintegrität, damit nicht zwei User gleichzeitig eine Datei umbennen, löschen usw. können bzw. klar geregelt ist, wer vorrang hat.

Port **21** - TCP

Layer 7 (Application Layer) - OSI Modell

Created 2022-03-07 08:53:36 UTC by Joshua Lieder
Updated 2022-04-01 10:54:49 UTC