Fachhochschule
Dortmund
University of Applied Sciences and Arts

Communications and Computer Networks

Summer Term 2023

Prof. Dr. Daniel Spiekermann
Faculty of Computer Science

Fachhochschule Dortmund
University of Applied Sciences and Arts

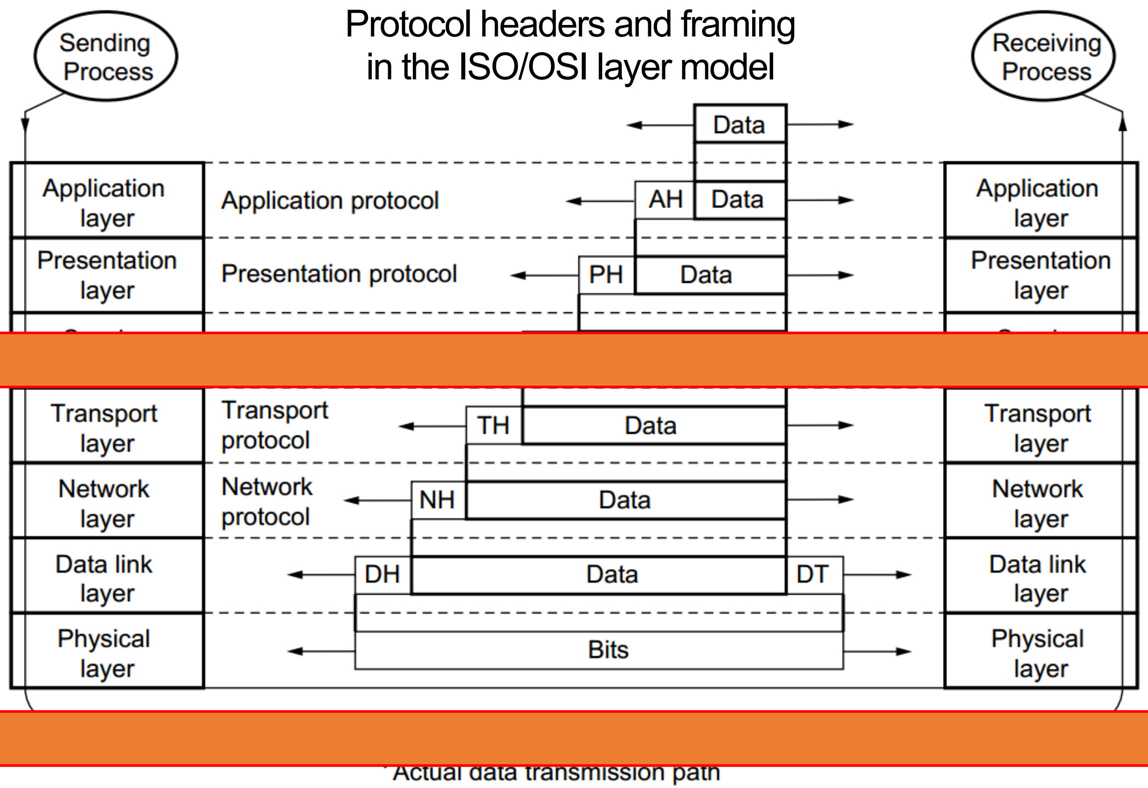
Recap of last lecture (1/8)

- You know the User Datagram Protocol (UDP) and can name its header structure
- You know the Transmission Control Protocol (TCP) and can name its header structure
- You know the principle of and can name selected port
- You know the tasks of TCP and can explain the communication process, including setting up and disconnecting the connection.
- You know the structure of the TCP header and can explain the fields.
- You can explain the principle of TCP window management.
- You are familiar with the Bandwidth-Delay product and can explain its importance for overload control using window control.
- You know the methods fast-retransmit and fast-recovery and can explain them.
- You know the QUIC protocol

Layer	
7	Application
6	Presentation
5	Session
4	**Transport**
3	Network
2	Data Link
1	Physical

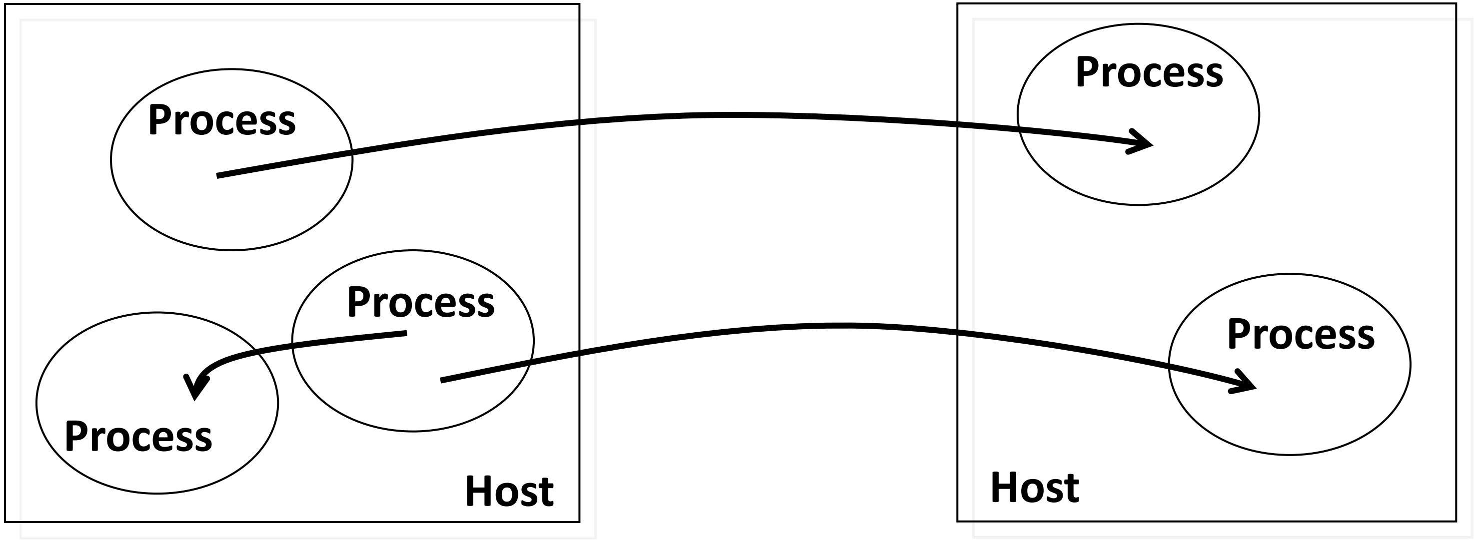
Recap of last lecture (2/8)

Discussion till now:



Recap of last lecture (3/8)

The ultimate goal of the communication mechanism between computers is that a **process** can communicate with another process that runs on another machine (or on the same one) in a transparent way.

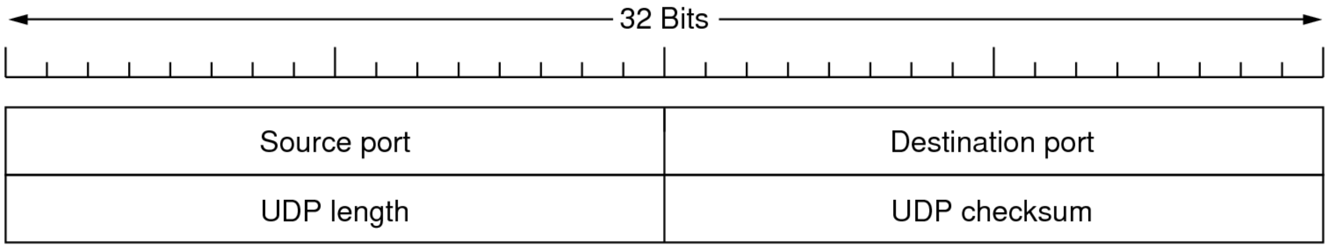


Fachhochschule Dortmund
University of Applied Sciences and Arts

Recap of last lecture (4/8)

Well known service	Reserved Port Number
TELNET	23
FTP	20, 21
DNS	53
HTTP	80
HTTPS	443
SMTP	25
POP3	110
IMAP4	143
IMAP4 over TLS/SSL	993
SSH	22
SNMP	161, 162

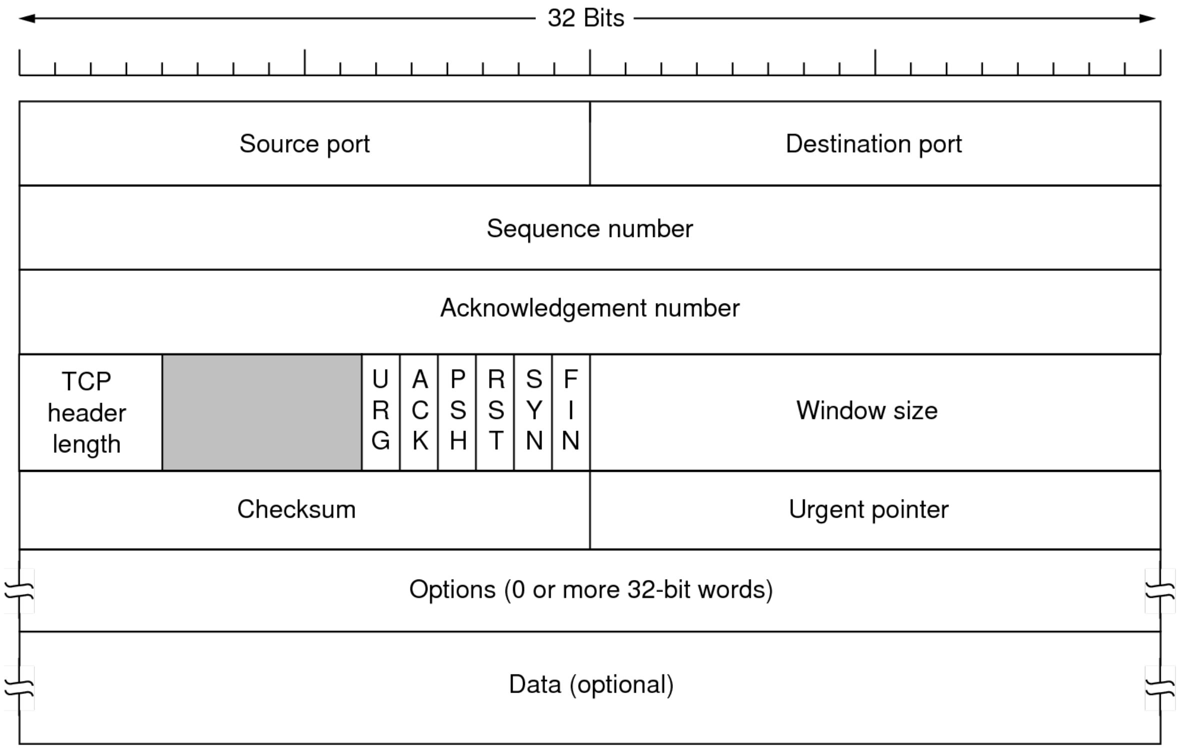
Recap of last lecture (5/8)



- **Source Port**: Port number of the sender
- **Destination Port**: Port number of the recipient
- **UDP Length**: Length of the datagram including header
- **UDP Checksum**: Checksum including pseudo-header

The use of Checksum is optional and is used to check the integrity of the header and the packet data. The checksum is calculated by adding a pseudo-header that contains the source and destination address, as well as the protocol number and the UDP packet length, in order to be able to verify without a doubt that the packet reached its correct recipient.

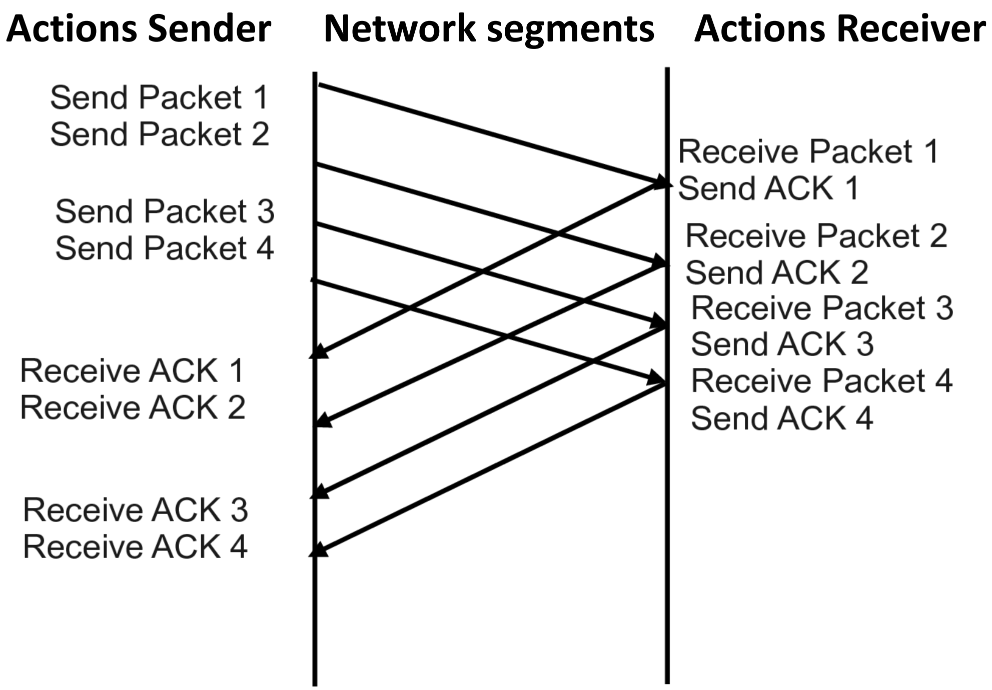
Recap of last lecture (6/8)



Fachhochschule Dortmund
University of Applied Sciences and Arts

Recap of last lecture (7/8)

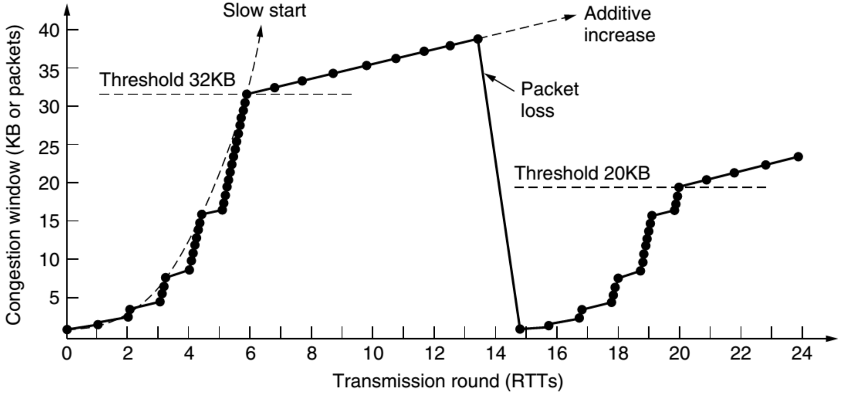
Sliding Window Protocol



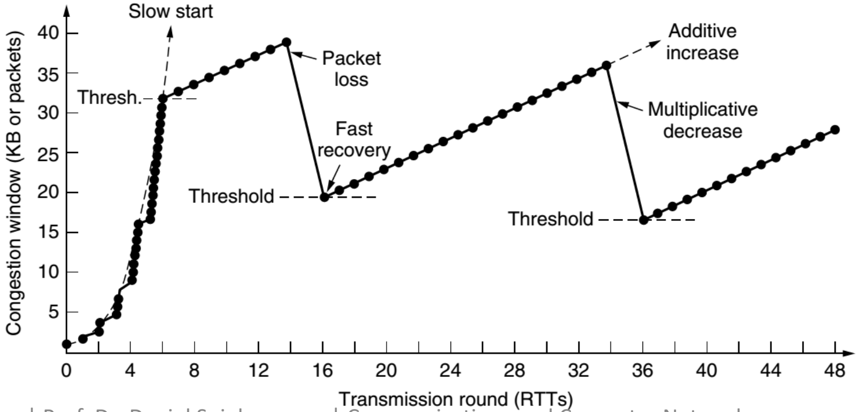
Fachhochschule Dortmund
University of Applied Sciences and Arts

Recap of last lecture (8/8)

- Slow Start &
- Fast Retransmit



- Slow Start &
- Fast Recovery



Fachhochschule Dortmund
University of Applied Sciences and Arts

Layer 7 - Application Layer

Prof. Dr. Daniel Spiekermann
Faculty of Computer Science

Layer 7 – Application layer

- The last link between application and transmission medium is the application layer.
- **Highest layer** of the OSI-stack
- It is the only layer that has an **interface to the application process.**
- The application process itself is "outside" the scope of the layer model.
- Relevant protocols:
  - DNS
  - DHCP
  - *HTTP*
  - *SMTP*
  - ...

Content Overview

- You know the task of DHCP
- You can describe the process and the involved DHCP messages
- You know the DHCPv6 protocol
- You understand the difference between stateful and stateless addresses
- You know DNS and the process of name resolution
- You understand the hierarchy of names, TLDs and SLDs as well as FQDNs
- You know HTTP and HTTPS
- You understand the request response communication and the interaction of application and protocol

Fachhochschule Dortmund
University of Applied Sciences and Arts

Application Layer - DHCP

Prof. Dr. Daniel Spiekermann
Faculty of Computer Science

Static ip address assignment

```
$ ip address add 192.168.11.1/24 dev eth0

$ ifconfig eth0 192.168.11.1 netmask 255.255.255.0

$ cat /etc/network/interfaces
iface eth0 inet static
address 192.168.11.1
netmask 255.255.255.0
gateway 192.168.11.254
```

Dynamic Host Configuration Protocol (RFC 2131)

- DHCP (Dynamic Host Configuration Protocol) enables the dynamic assignment of an IP address (IPv4) and other configuration parameters to computers on a network using a DHCP server.
- Using DHCP, a server can automatically assign parameters such as IP address, netmask, gate-way and DNS server when starting a new computer
- DHCP is based on the BOOTP protocol, which can be used to implement driveless workstations that first get an IP address from the BOOTP server, and then reload a bootable operating system from the network, with which they then boot.
- The technology **eliminates the need for individually configuring network devices manually**, and consists of two network components, a centrally installed network DHCP **server** and **client** instances of the protocol stack on each computer or device. When connected to the network, and periodically thereafter, a client requests a set of parameters from the DHCP server using the DHCP protocol.
- DHCP services exist for networks running Internet Protocol version 4 (**IPv4**), as well as version 6 (**IPv6**). The IPv6 version of the DHCP protocol is commonly called DHCPv6.

DHCP Operation

- The DHCP employs a **connectionless** service model, using the User Datagram Protocol (**UDP**). It is implemented with two UDP port numbers for its operations which are the same as for the bootstrap protocol (BOOTP). UDP port number **67** is the destination port of a **server**, and **UDP** port number **68** is used by the client.
- DHCP operations fall into four phases: server **discovery**, IP lease **offer**, IP lease **request**, and IP lease **acknowledgement**. These stages are often abbreviated as **DORA** for discovery, offer, request, and acknowledgement.
- The DHCP operation begins with clients broadcasting a request. If the client and server are in different Broadcast Domains, a DHCP Helper or **DHCP Relay Agent** may be used. Clients requesting renewal of an existing lease may communicate directly via UDP unicast, since the client already has an established IP address at that point.

DHCP-Server allocation

- Listens on port 67 UDP
- Provides at least a range of IP-addresses
- **Dynamic allocation**
  A network administrator reserves a range of IP addresses for DHCP, and each DHCP client on the LAN is configured to request an IP address from the DHCP server during network initialization.

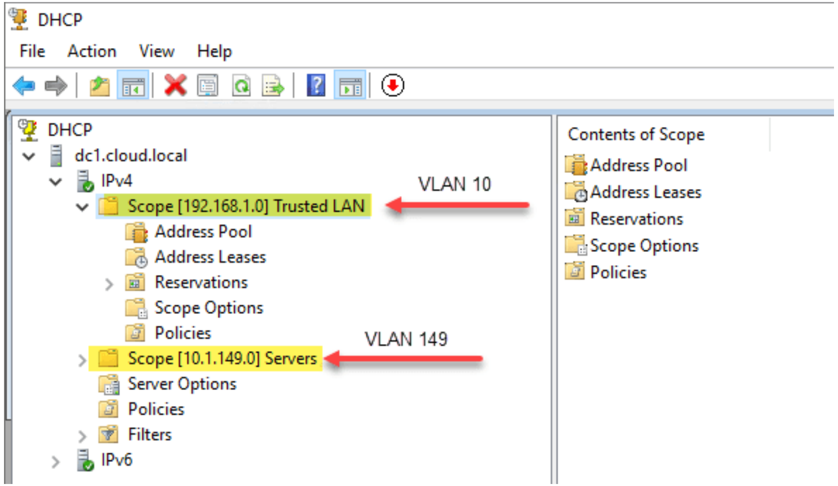
```
subnet 192.168.1.0 netmask 255.255.255.0 {
range 192.168.1.10 192.168.1.100;
range 192.168.1.150 192.168.1.200;
}
```

- **Manual allocation**
  This method is also variously called static DHCP allocation, fixed address allocation, reservation, and MAC/IP address binding. An administrator maps a unique identifier (a client id or MAC address) for each client to an IP address, which is offered to the requesting client.

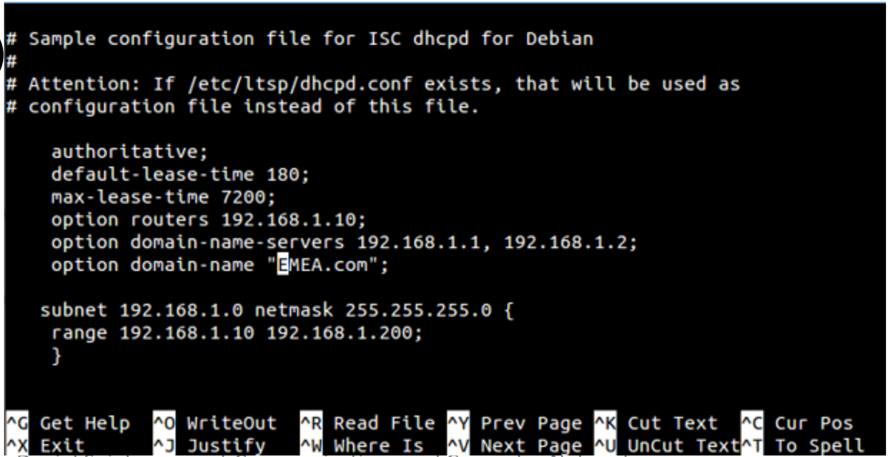
```
# always set IP to 10.0.3.1 for web1 vm
10.0.3.1,web1
# always set IP to 10.0.5.12 for db12 vm based on mac
00:16:ee:62:ff:90,10.0.5.12,db12
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCP Server



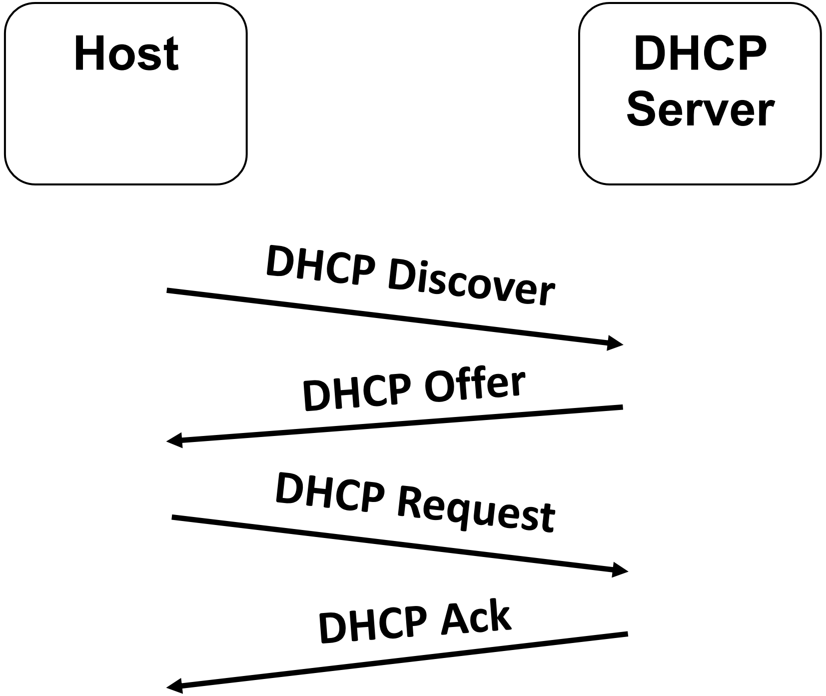
- Typically every DSL-Router provides a DHCP-server
- Dedicated DHCP-server for OS exist
  - Windows Server
  - dnsmasq
  - isc-dhcpd (deprecated since 2022)
  - Kea



```
# Sample configuration file for ISC dhcpd for Debian
#
# Attention: If /etc/ltsp/dhcpd.conf exists, that will be used as
# configuration file instead of this file.

    authoritative;
    default-lease-time 180;
    max-lease-time 7200;
    option routers 192.168.1.10;
    option domain-name-servers 192.168.1.1, 192.168.1.2;
    option domain-name "EMEA.com";

   subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.10 192.168.1.200;
    }

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

DHCP DORA

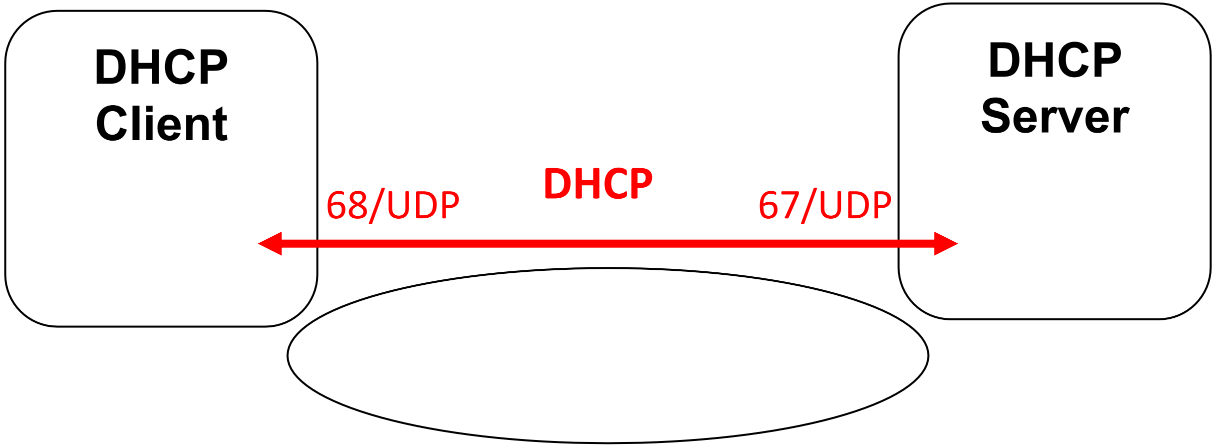
- **DHCP Discover**
  broadcast to locate servers
- **DHCP Offer**
  each server offers configuration parameters
- **DHCP Request**
  request offered parameters from one server
- **DHCP Ack**
  assignment confirmation



Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCP and Wireshark

Source	Destination	Protocol	Length	Info
0.0.0.0	255.255.255.255	DHCP	618	DHCP Discover
192.168.0.1	255.255.255.255	DHCP	342	DHCP Offer
0.0.0.0	255.255.255.255	DHCP	618	DHCP Request
192.168.0.1	255.255.255.255	DHCP	342	DHCP ACK



DHCP structure

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="8">OP</td><td colspan="8">HTYPE</td><td colspan="8">HLEN</td><td colspan="8">HOPS</td></tr>
<tr><td colspan="32">XID</td></tr>
<tr><td colspan="16">SECS</td><td colspan="16">FLAGS</td></tr>
<tr><td colspan="32">CIADDR</td></tr>
<tr><td colspan="32">YIADDR</td></tr>
<tr><td colspan="32">SIADDR</td></tr>
<tr><td colspan="32">GIADDR</td></tr>
<tr><td colspan="32">CHADDR</td></tr>
<tr><td colspan="32">SNAME</td></tr>
<tr><td colspan="32">FILE</td></tr>
<tr><td colspan="32">OPTIONS</td></tr>
</table>

Fachhochschule Dortmund
University of Applied Sciences and Arts

Wireshark and DHCP

```
> Ethernet II, Src: cc:01:0a:c4:00:00 (cc:01:0a:c4:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 255.255.255.255
> User Datagram Protocol, Src Port: 67, Dst Port: 68
v Dynamic Host Configuration Protocol (Offer)
    Message type: Boot Reply (2)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x0000155c
    Seconds elapsed: 0
  > Bootp flags: 0x8000, Broadcast flag (Broadcast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 192.168.0.3
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: cc:00:0a:c4:00:00 (cc:00:0a:c4:00:00)
```

Wireshark and DHCP

Broadcast

Unique MAC from sender

```
> Ethernet II, Src: cc:01:0a:c4:00:00 (cc:01:0a:c4:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 255.255.255.255
> User Datagram Protocol, Src Port: 67, Dst Port: 68
v Dynamic Host Configuration Protocol (Offer)
    Message type: Boot Reply (2)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x0000155c
    Seconds elapsed: 0
  > Bootp flags: 0x8000, Broadcast flag (Broadcast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 192.168.0.3
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: cc:00:0a:c4:00:00 (cc:00:0a:c4:00:00)
```

Port 67 und 68

Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCP structure

- **OP**: opcode 1 = REQUEST, 2 = REPLY
- HTYPE: Hardware address type (01 p/MAC Addresses)
- HLEN: Hardware address length (06 p/MAC Addresses)
- HOPS: Client sets to zero, optionally used by relay agents
- **XID**: Transaction ID
- SECS: seconds elapsed since client began addr acquisition
- FLAGS: BROADCAST flag
- **CIADDR**: Client IP address
- **YIADDR**: 'your' (client) IP address, the offer of the server
- **SIADDR**: IP address of server
- **GIADDR**: Relay agent IP address
- **CHADDR**: Client hardware address
- SNAME: Optional server host name

Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCP Parameters and Options

- DHCP provides assignment of further parameters (so-called options)
  - Subnet Mask (1)
  - Default Gateway (3)
  - Time-Server (4)
  - DNS Server (6)
  - Hostname (12)
  - End (255)

```
> Option: (12) Host Name
> Option: (1) Subnet Mask (255.255.255.0)
v Option: (3) Router
    Length: 4
    Router: 192.168.0.1
v Option: (6) Domain Name Server
    Length: 8
    Domain Name Server: 192.168.0.1
    Domain Name Server: 192.168.1.1
> Option: (255) End
```

DHCP Lease

- **Different times in DHCP ACK**
- LEASE time (Option 51)
  - Time period for which the assignment is valid.
- Renewal Time (50% of the length of the lease)
  - This option specifies the time interval from address assignment until the client transitions to the RENEWING state. Client sends *DHCPREQUEST*
- Rebinding Time (87.5% (7/8ths) of the length of the lease)
  - This option specifies the time interval from address assignment until the client transitions to the REBINDING state. When the server does not reply to the DHCPREQUEST, client tries to connect to any DHCP-server in the network

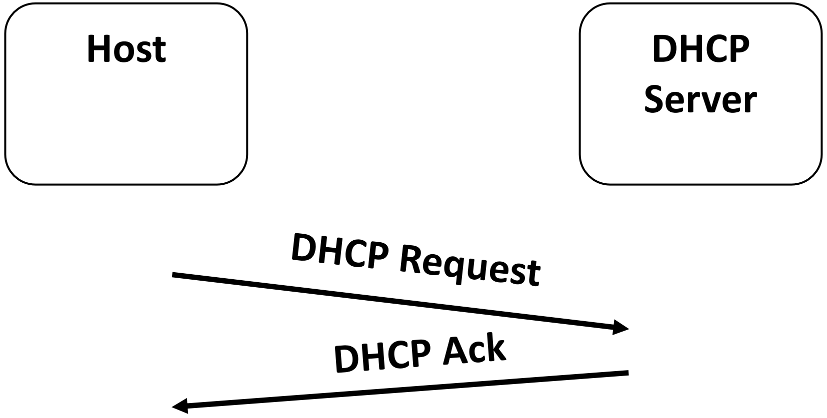
```
Option: (51) IP Address Lease Time
    Length: 4
    IP Address Lease Time: (60s) 1 minute
Option: (58) Renewal Time Value
    Length: 4
    Renewal Time Value: (30s) 30 seconds
Option: (59) Rebinding Time Value
    Length: 4
    Rebinding Time Value: (52s) 52 seconds
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCP Request

DHCP Request

- Send to the server after renewal timer reached 0
- Send directly to the server, no broadcast
- Server answers with DHCP ACK, if requested IP-address is still available for the host



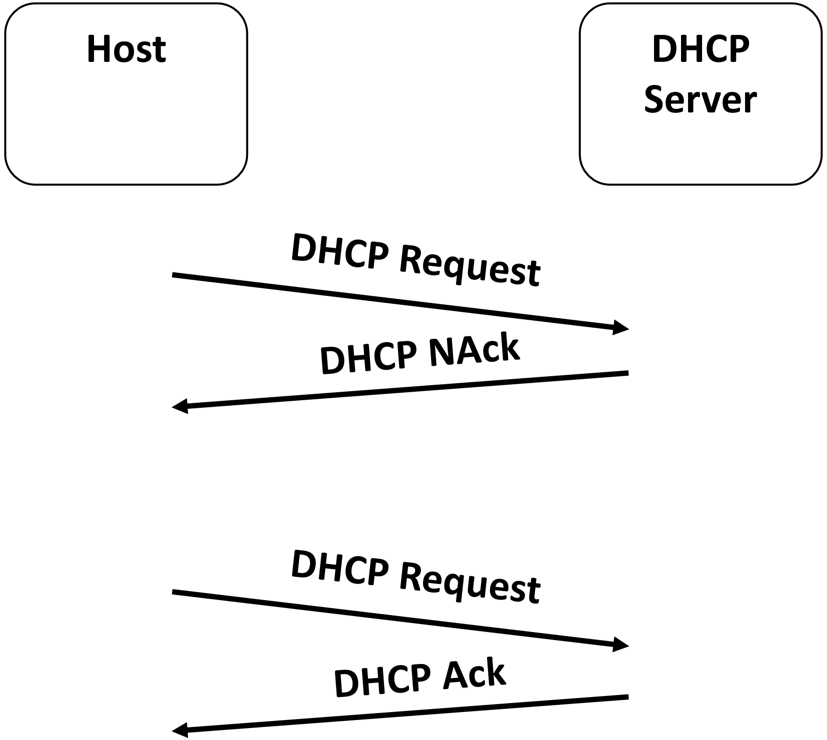
DHCP NACK

DHCP NACK

- Typical case of network change.

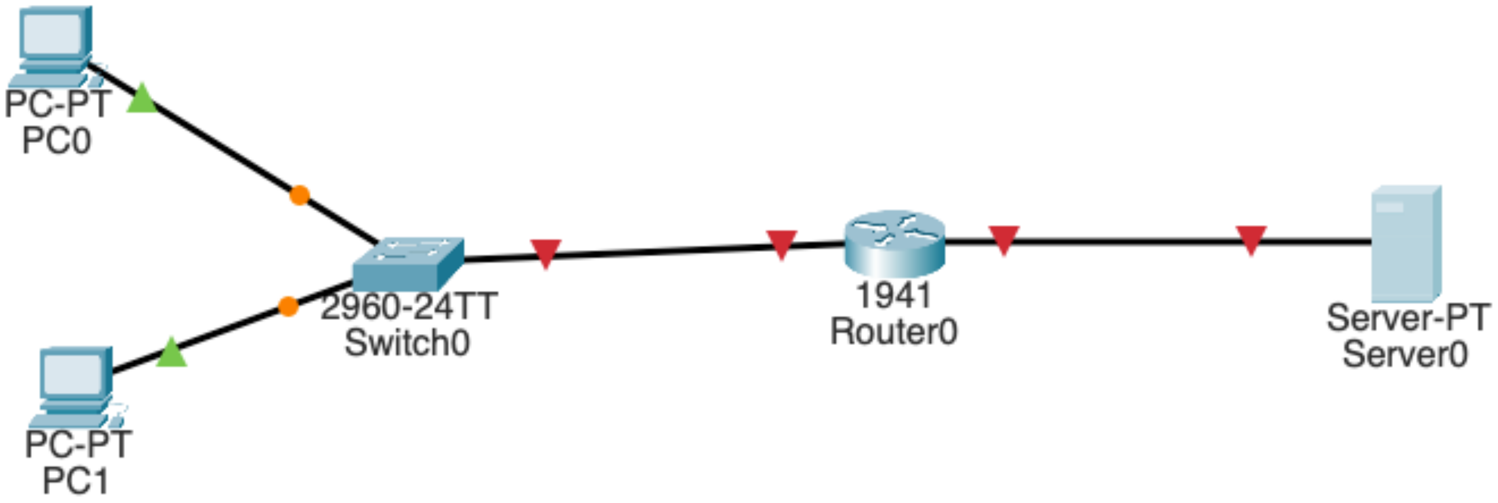
The host already has an IP and when reconnecting to a network, it requests the renewal of the Lease with a DHCP Request, but the IP is no longer valid in that network.

Host sends a new DHCP Request to the server



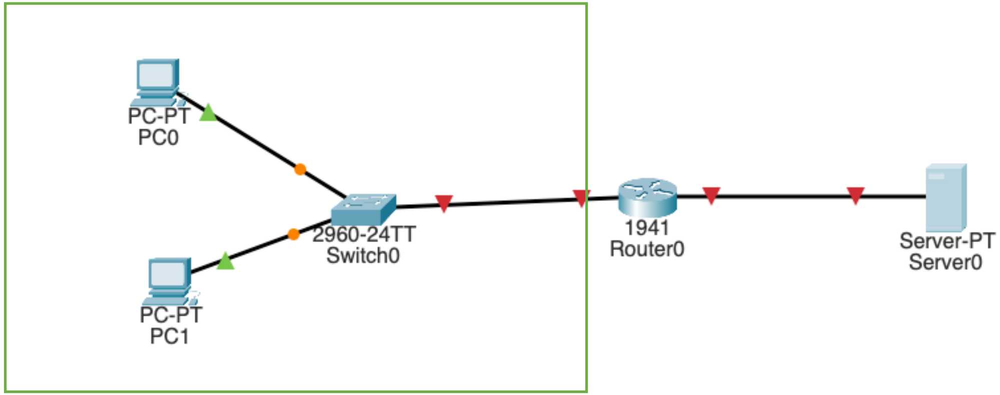
Fachhochschule Dortmund
University of Applied Sciences and Arts

A “new” problem



DHCP Relaying

- The DHCP client broadcasts on the local link, **the client's broadcast can only be received on its own subnet**.
- **DHCP relay agents** can be installed on these subnets. A DHCP relay agent runs on a network device, capable of routing between the client's subnet and the subnet of the DHCP server.



DHCP Relaying

- The DHCP client broadcasts on the local link, **the client's broadcast can only be received on its own subnet**.
- **DHCP relay agents** can be installed on these subnets. A DHCP relay agent runs on a network device, capable of routing between the client's subnet and the subnet of the DHCP server.
- The DHCP server uses the GIADDR-value to determine the subnet, and subsequently the corresponding address pool, from which to allocate an IP address. When the DHCP server replies to the client, it sends the reply to the GIADDR-address, again using unicast.
- The relay agent then retransmits the response on the local network, using unicast (in most cases) to the newly reserved IP address, in an ethernet frame directed to the client's MAC address.
- The communication between the relay agent and the DHCP server typically uses both a source and destination **UDP** port of **67**.

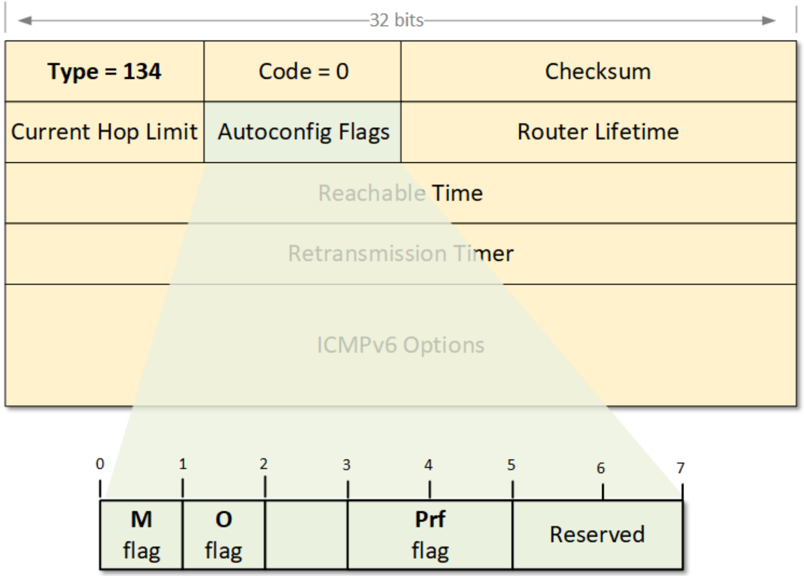
Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCPv6 (RFC 3315, now RFC 8415)

- Uses UDP port 546 (client) and 547 (server)
- DHCP server (and relay agents) listen on ff02::1:2 (Multicast) or ff05::1:3
- Client requests
- Stateless:
  - Prefix is assigned by RA
  - DHCPV6 provides network information like DNS server
- Stateful:
  - RA with flag “managed”
  - Full ip address configuration assigned by the server
- Server and client have a DUID (DHCP Unique Identifier)
  - DHCP servers use DUIDs to identify clients for the selection of configuration parameters and in the association of IAs with clients.
  - DHCP clients use DUIDs to identify a server in messages where a server needs to be identified.
- LLA is still calculated by the client itself

DHCPv6 stateful with RA

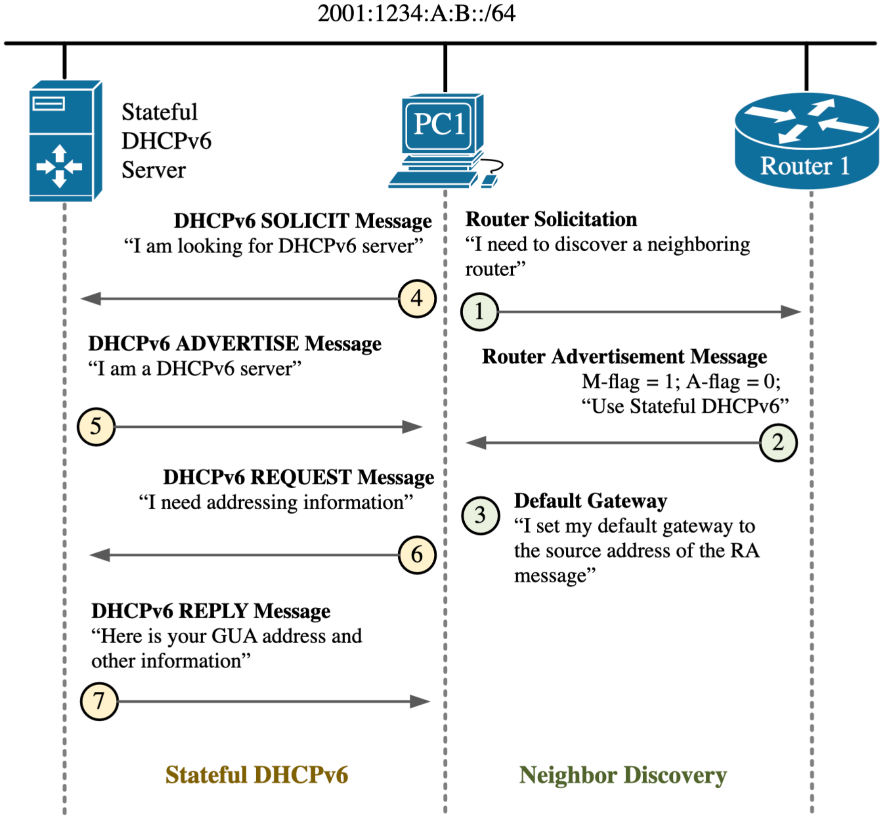
- Needs specific configuration of the router
  - set M-flag = 1
- ICMPv6 Type134 (RA) is send after RS
- Client is informed to get all information from a DHCPv6 server




```
Internet Control Message Protocol v6
  Type: Router Advertisement (134)
  Code: 0
  Checksum: 0x9a04 [correct]
  [Checksum Status: Good]
  Cur hop limit: 64
  Flags: 0xc0, Managed address configuration, Other configuration,
    1... .... = Managed address configuration: Set
    .1.. .... = Other configuration: Set
    ..0. .... = Home Agent: Not set
    ...0 0... = Prf (Default Router Preference): Medium (0)
    .... .0.. = Proxy: Not set
    .... ..0. = Reserved: 0
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

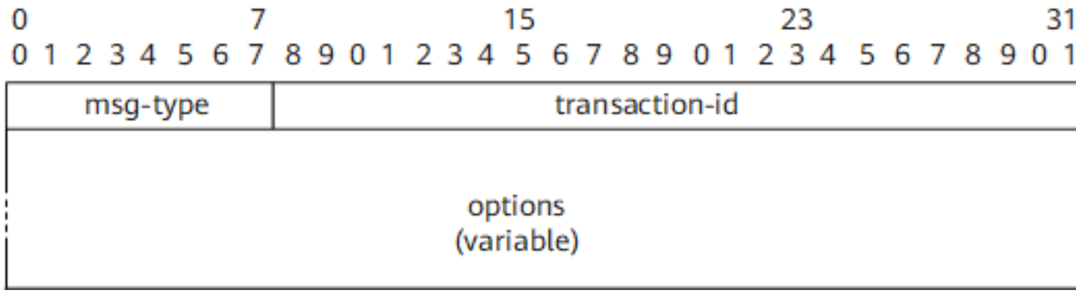
DHCPv6 stateful with RA



Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCPv6 Protocol format

- Msg-type: Message Type
- Transaction-id: Identifies trans-action between client and server
- Options: Indicates an option field
  - Option-code
    - 1: Client-ID
    - 2: Server-ID
    - 3: Identity Association for Non-temporary Addresses
    - 18: IID
  - Option-length
  - Option-data



DHCPv6 Message Types

13 different msg-types

Type	Packet	DHCPv4 mapping
1	Solicit	Discover
2	Advertise	Offer
3	Request	Request
4	Confirm	-
5	Renew	Request
6	Rebind	Request
7	Reply	ACK/NACK

Fachhochschule Dortmund
University of Applied Sciences and Arts

DHCPv6 and Wireshark

1.) Request to all servers or relay agents

2.) Answer of a server intended, but LLA not known, so NS send

fe80::a00:27ff:fefe:8f95	ff02::1:2	DHCPv6	114	UDP	Solicit XID: 0x100874 CID: 000100011c39cf8
fe80::a00:27ff:fed4:10bb	ff02::1:fffe:8f95	ICMPv6	86	ICMPv6	Neighbor Solicitation for fe80::a00:27ff:f
fe80::a00:27ff:fefe:8f95	fe80::a00:27ff:fed4:10bb	ICMPv6	86	ICMPv6	Neighbor Advertisement fe80::a00:27ff:fefe
fe80::a00:27ff:fed4:10bb	fe80::a00:27ff:fefe:8f95	DHCPv6	147	UDP	Advertise XID: 0x100874 CID: 000100011c39c
fe80::a00:27ff:fed4:10bb	ff02::16	ICMPv6	110	IPv6 Hop…	Multicast Listener Report Message v2
fe80::a00:27ff:fefe:8f95	ff02::1:2	DHCPv6	161	UDP	Request XID: 0x49174e CID: 000100011c39cf8
fe80::a00:27ff:fed4:10bb	fe80::a00:27ff:fefe:8f95	DHCPv6	147	UDP	Reply XID: 0x49174e CID: 000100011c39cf880

3.) Client answers with NA with its LLA

4.) Server sends DHCPv6 on port 546 to LLA with the requested information

```
Ethernet II, Src: PcsCompu_d4:10:bb (08:00:27:d4:10:bb), Dst: PcsCompu_fe:8f:95 (08:00:27:fe:
Internet Protocol Version 6, Src: fe80::a00:27ff:fed4:10bb, Dst: fe80::a00:27ff:fefe:8f95
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
   Message type: Advertise (2)
   Transaction ID: 0x100874
 ∨ Identity Association for Prefix Delegation
      Option: Identity Association for Prefix Delegation (25)
      Length: 41
      IAID: 27fe8f95
      T1: 0
      T2: 0
    ∨ IA Prefix
         Option: IA Prefix (26)
         Length: 25
         Preferred lifetime: 4500
         Valid lifetime: 7200
         Prefix length: 64
         Prefix address: 2001:0:0:fe00::
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

Application Layer - DNS

Prof. Dr. Daniel Spiekermann
Faculty of Computer Science

Fachhochschule Dortmund
University of Applied Sciences and Arts

The Domain Name System (RFCs 1034, 1035, 1591)

You want to communicate with www.fh-dortmund.de

What do we need?

Fachhochschule Dortmund
University of Applied Sciences and Arts

The Domain Name System (RFCs 1034, 1035, 1591)

You want to communicate with www.fh-dortmund.de

We have learned, that communication „uses“ ip-addresses

Fachhochschule Dortmund
University of Applied Sciences and Arts

The Domain Name System (RFCs 1034, 1035, 1591)

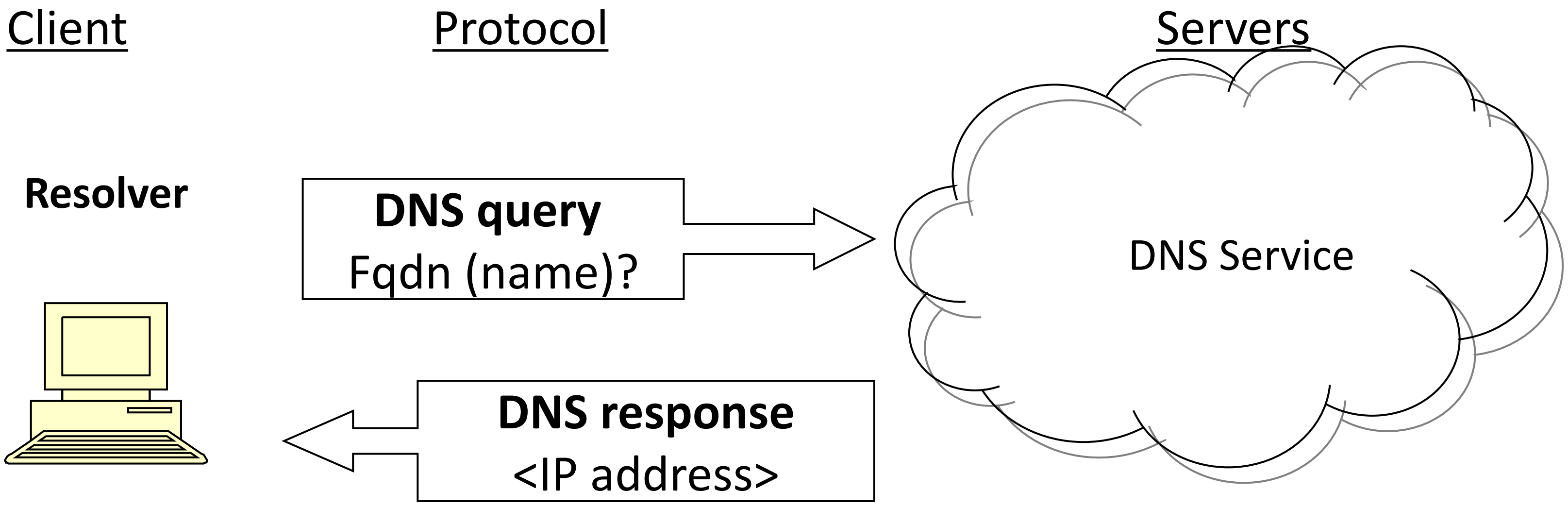
You want to communicate with www.fh-dortmund.de

We have learned, that communication „uses“ ip-addresses

So, what is the current ipv4 or ipv6 address of www.fh-dortmund.de?

Fachhochschule Dortmund
University of Applied Sciences and Arts

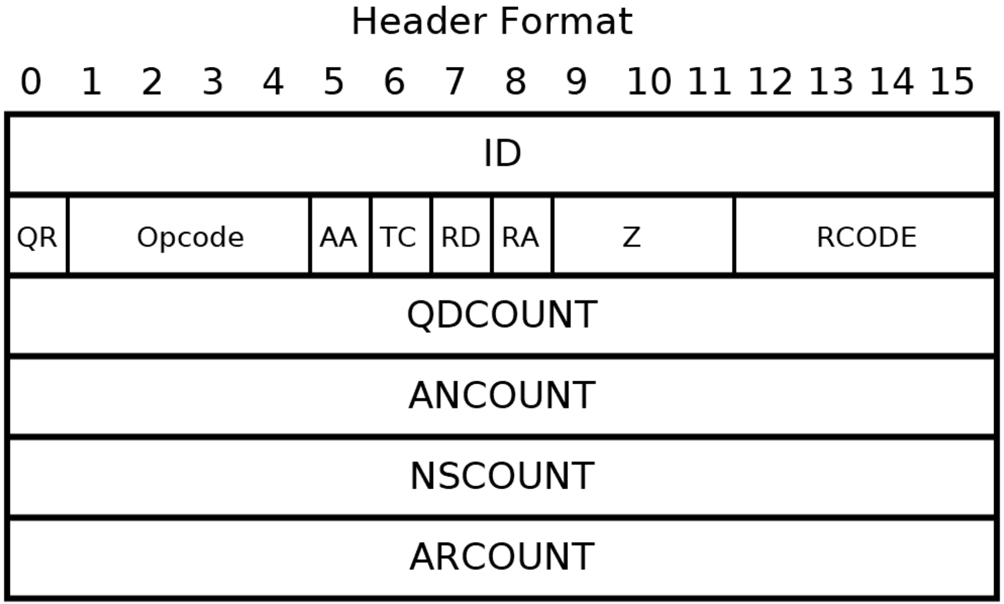
DNS lookups



Protocol format

ID: 16-bit value that is used to identify the DNS message

- **QR (query/response):**
  0 = request, 1 = response type.
- **opcode:** type of query carried by a message. This field value is repeated in the response.
- **AA:** Authoritative Answer.
- **TC:** Truncation
- **RD:** Recursion Desired
- **RA:** Recursion Available
- **Zero:** reserved, set to 0.
- **rCode:** Response Code. query was answered successfully or not.



Fachhochschule Dortmund
University of Applied Sciences and Arts

Wireshark - DNS Request

```
User Datagram Protocol, Src Port: 55719, Dst Port: 53
Domain Name System (query)
    Transaction ID: 0x7cf4
  ⌄ Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ⌄ Queries
    ⌄ www.neverssl.com: type A, class IN
        Name: www.neverssl.com
        [Name Length: 16]
        [Label Count: 3]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
    [Response In: 116]
```

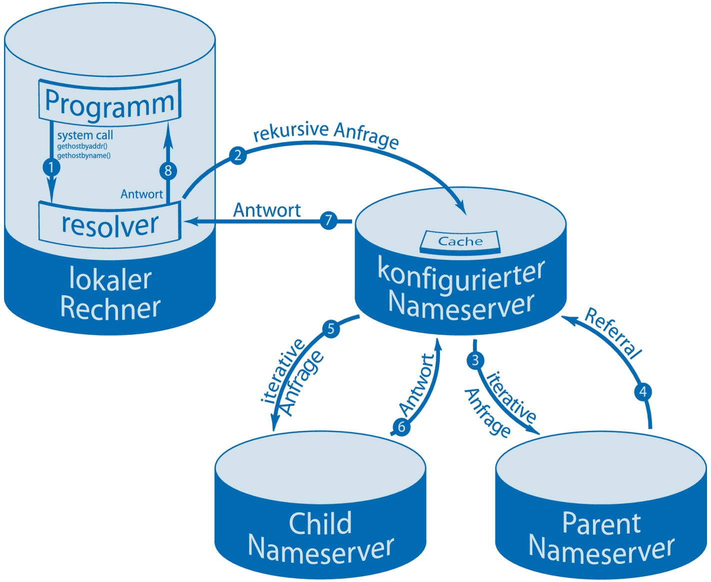
Fachhochschule
Dortmund
University of Applied

Wireshark – DNS response

```
Domain Name System (response)
    Transaction ID: 0x7cf4
  ⌄ Flags: 0x8180 Standard query response, No error
        1... .... .... .... = Response: Message is a response
        .000 0... .... .... = Opcode: Standard query (0)
        .... .0.. .... .... = Authoritative: Server is not an authority for domain
        .... ..0. .... .... = Truncated: Message is not truncated
        .... ...1 .... .... = Recursion desired: Do query recursively
        .... .... 1... .... = Recursion available: Server can do recursive queries
        .... .... .0.. .... = Z: reserved (0)
        .... .... ..0. .... = Answer authenticated: Answer/authority portion was not a
        .... .... ...0 .... = Non-authenticated data: Unacceptable
        .... .... .... 0000 = Reply code: No error (0)
    Questions: 1
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 0
  › Queries
  ⌄ Answers
      ⌄ www.neverssl.com: type A, class IN, addr 34.223.124.45
            Name: www.neverssl.com
            Type: A (Host Address) (1)
            Class: IN (0x0001)
            Time to live: 49 (49 seconds)
            Data length: 4
            Address: 34.223.124.45
    [Request In: 115]
    [Time: 0.256504000 seconds]
```

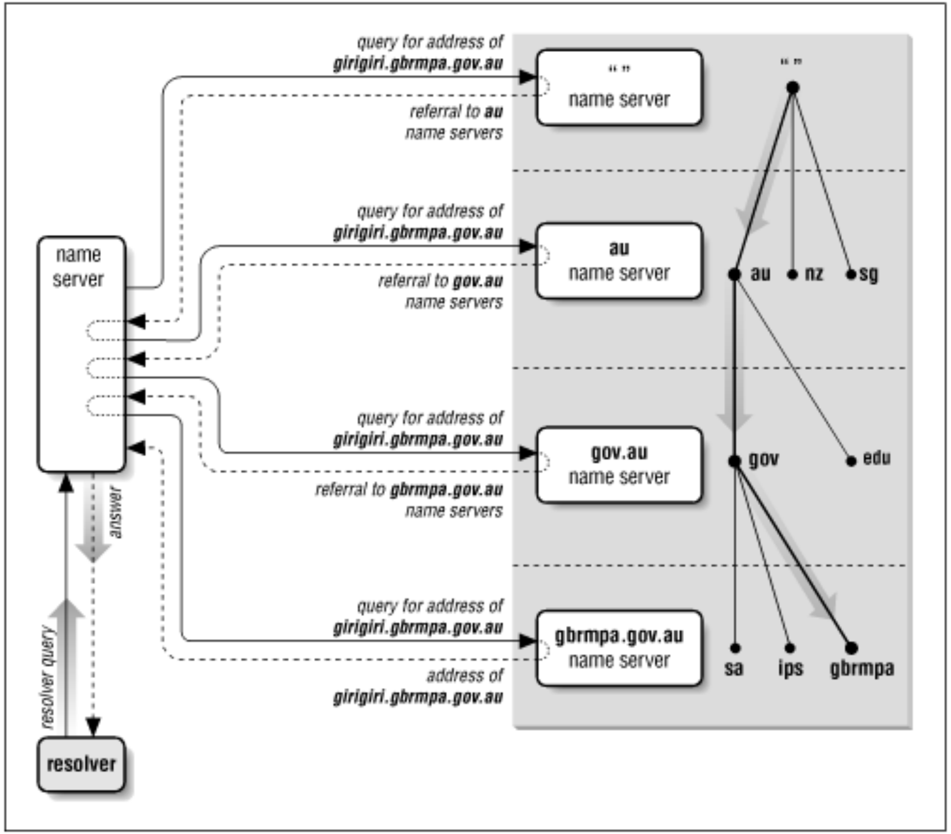
Types of DNS requests and responses

- Authoritative: A name server responds with the information from its local zone file.
- Recursive: The name server itself sends requests to other nameservers before responding. (The requester does not see these requests.)
- Iterative: The requested name server does not send a direct response but points to another name server (forwarding). Root servers answer requests only iteratively.

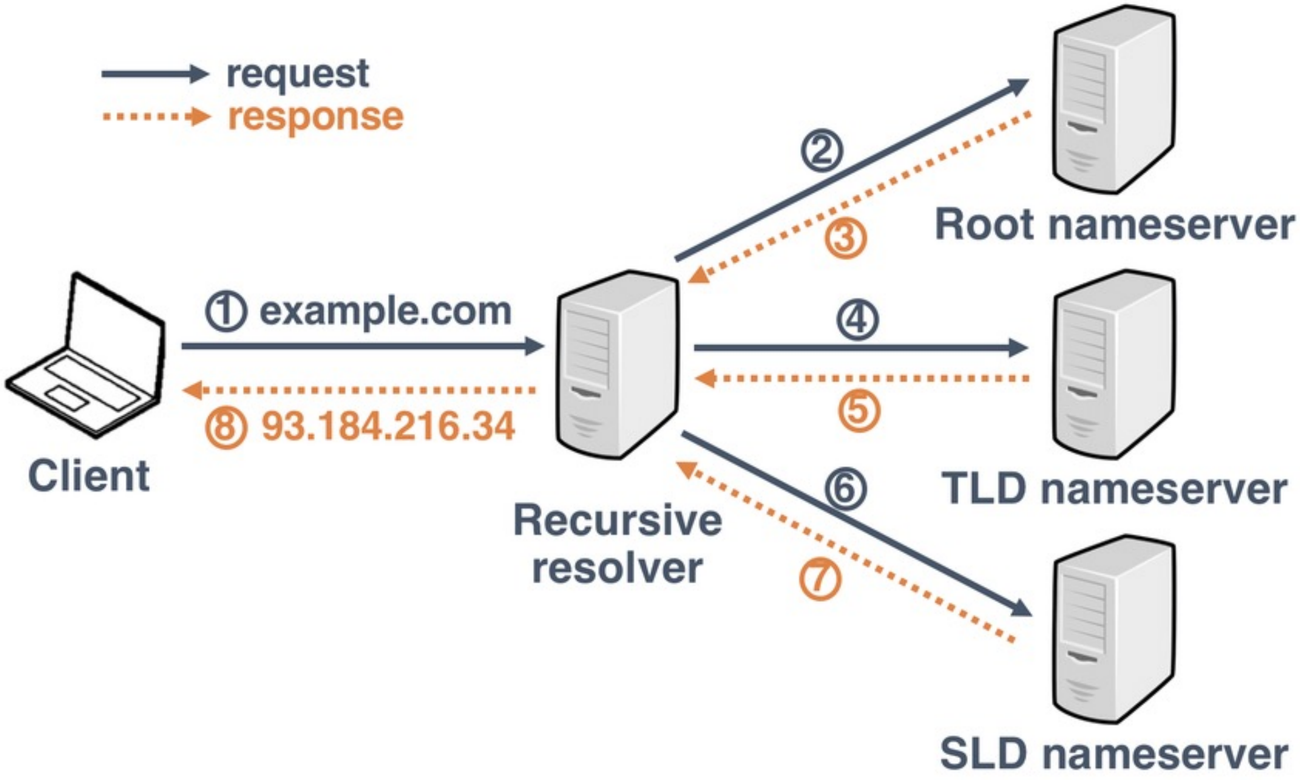


Fachhochschule
Dortmund
University of Applied Sciences and Arts

Iterative name resolution



Recursive

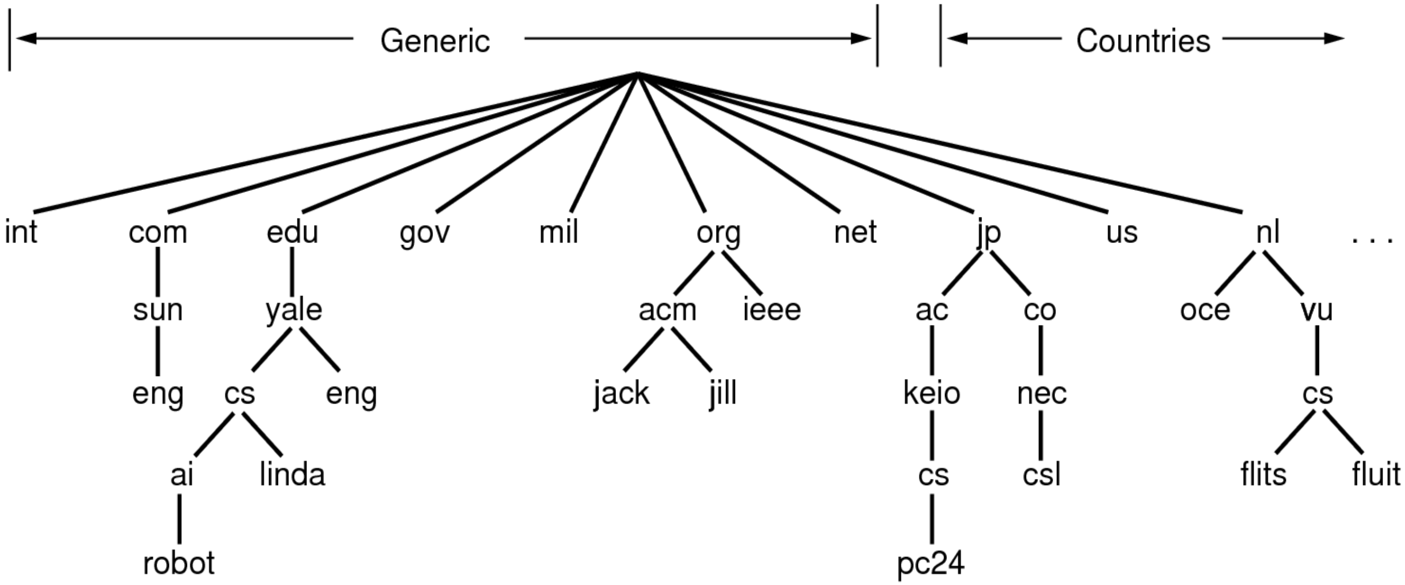

request
response
① example.com
⑧ 93.184.216.34
Client
Recursive
resolver
②
③
Root nameserver
④
⑤
TLD nameserver
⑥
⑦
SLD nameserver

DNS overview

- It is a **single distributed database**
- The database is **distributed** among many individual servers, which form a federation of nodes that collectively provide the DNS service (the DNS service is implemented in a distributed, not centralized manner)
- **Registrations**, cancellations and modifications to the data are also done decentralized. The data set is divided into parts called **zones**. For each zone, an **administrative entity** is designated that has permission (**authority**) to modify data contained in the zone.
- The DNS data is structured according to a **hierarchical** model (tree) to organize the data

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS hierarchy



DNS services

- Name resolution
  - Name -> IP-address
  - www.fh-dortmund.de -> 193.25.16.26
  - *and vice versa*
- Load Balancing
  - A name can map to multiple hosts thus multiple addresses
  - DNS server returns all addresses but rotates ordering
- Mail server addressing
  - Special mail server addressing apart from normal name resolution

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS Resource Records

Field	Description	Length (octets)
NAME	**fully qualified domain name** of the node to which this record pertains	Variable
TYPE	Type of RR in numeric form (e.g., 1 for A)	2
CLASS	Class code (set to **IN** for Internet)	2
TTL	**Time To Live** Seconds that the RR could stays valid in cache	4
RDLENGTH	Length of RDATA field (specified in octets)	2
RDATA	Additional **RR-specific data**	Variable

DNS RR Type

▪ A	Type: 1	RData: IPv4 address
▪ NS	Type: 2	RData: FQDN Name Server
▪ CNAME	Type: 5	RData: FQDN alias
▪ SOA	Type: 6	RData: Start Of Authority
▪ PTR	Type: 12	RData: FQDN reverse lookup
▪ MX	Type: 15	RData: FQDN mail exchange server
▪ TXT	Type: 16	RData: Text
▪ AAAA	Type: 28	RData: IPv6 address
▪ DNSKEY	Type: 48	RData: PublicKey
▪ HTTPS	Type: 65	RData: Improved access to https resources

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS RR A

Resolves a hostname to an IPv4-address

$ nslookup
> set query=**A**
> **www.fh-dortmund.de**
Server: 192.168.1.1
Address: 192.168.1.1#53

Non-authoritative answer:
Name: **www.fh-dortmund.de**
Address: **193.25.16.26**

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS RR AAAA

Resolves a hostname to an IPv6-address

$ nslookup
> set query=**AAAA**
> www.google.de
Server: fe80::d487:62:9df4:e3dd%15
Address: fe80::d487:62:9df4:e3dd%15#53

Non-authoritative answer:
www.google.de has AAAA address 2a00:1450:4001:829::2003

DNS RR CNAME

- Maps one domain name (an alias) to another(the canonical name).

```
Answers
  www.wdr.de: type CNAME, class IN, cname www.wdr.de.v1.edgekey.net
    Name: www.wdr.de
    Type: CNAME (Canonical NAME for an alias) (5)
    Class: IN (0x0001)
    Time to live: 44 (44 seconds)
    Data length: 27
    CNAME: www.wdr.de.v1.edgekey.net
  www.wdr.de.v1.edgekey.net: type CNAME, class IN, cname e8381.e6.akamaiedge.net
    Name: www.wdr.de.v1.edgekey.net
    Type: CNAME (Canonical NAME for an alias) (5)
    Class: IN (0x0001)
    Time to live: 20547 (5 hours, 42 minutes, 27 seconds)
    Data length: 22
    CNAME: e8381.e6.akamaiedge.net
```

DNS RR MX

Resolves the IP-address of the responsible mail server for this domain

$ nslookup
> set query=**MX**
> **www.fh-dortmund.de**
Server: 192.168.1.1
Address: 192.168.1.1#53

Non-authoritative answer:
fh-dortmund.de mail exchanger = 0 fhdortmund-de0e.mail.protection.outlook.com.

DNS RR NS

Resolves the authoritative nameserver for this domain

$ nslookup
> set query=**NS**
> **fh-dortmund.de**
Server: 192.168.1.1
Address: 192.168.1.1#53

Non-authoritative answer:
fh-dortmund.de nameserver = ns2.inwx.de.
fh-dortmund.de nameserver = ns3.inwx.eu.
fh-dortmund.de nameserver = ns.inwx.de.

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS RR SOA

Provides administrative information about the zone

$ nslookup
> set query=**soa**
> **fh-dortmund.de**
Server: 192.168.1.1
Address: 192.168.1.1#53

Non-authoritative answer:
fh-dortmund.de
origin = ns.inwx.de
mail addr = hostmaster.inwx.de
serial = 2023060500
refresh = 1800
retry = 900
expire = 21600
minimum = 600

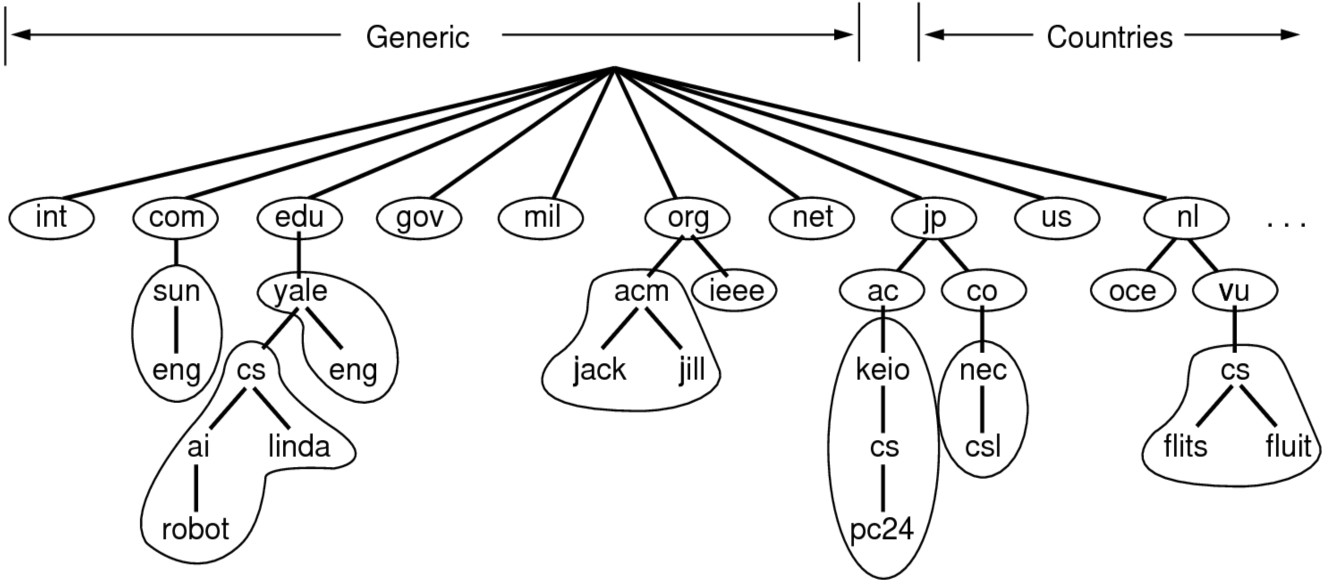
DNS Domains & Zones

- **Domain**
  the totality of descendant nodes of a certain node (a sub-tree)
- **Zone**
  a portion of a domain, administered by an administrative entity. The boundaries of the zones are set by how authority is partitioned and distributed over the distributed database.

- For administration, the (single) DNS tree is divided into **zones**.
  The administrative entity that has **authority** over a portion (zone) of the namespace can add, remove, or change labels within that zone.
  A zone begins at a certain node, and includes all nodes descended from it, except nodes belonging to sub-zones whose authority has previously been **delegated** to other entities.

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS Zones

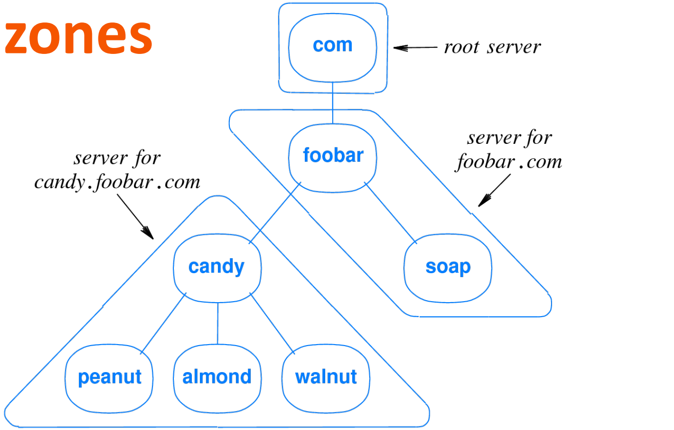


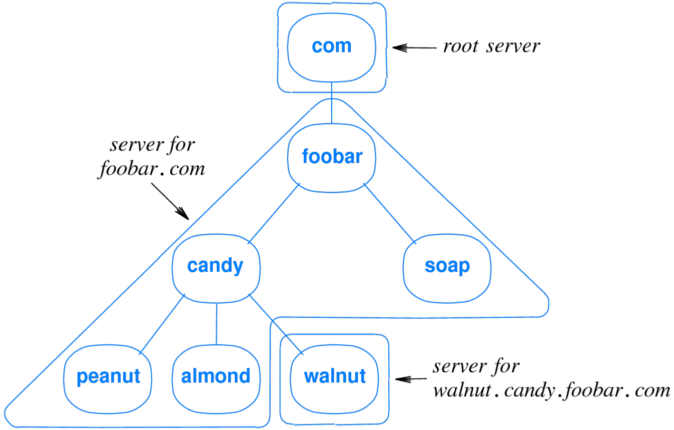
The illustrations show examples of dividing individual domains into zones.

Fachhochschule
Dortmund
University of Applied Sciences and Arts

Division of a domain in zones

- A single server cannot manage all entries in a domain
- Dividing the domain into zones
- The division into zones is the responsibility of the driver of the domain.
- In the upper picture, a DNS Server for foobar.com and another DNS server for candy.foobar.com are used.
- In the lower picture, a DNS server for foobar.com and another DNS Server for walnut.candy.foobar.com are used.
- Structuring the domain into zones is not visible from the outside.





Fachhochschule
Dortmund
University of Applied Sciences and Arts

DNS zone file and FQDN

- Stores all information related to the specified zone
  - Mapping between domain names and ip-addresses
  - Can store further information (DNSKEY, HTTPS)
- Text file format (defined in RFC 1034 and RFC 1035)
- Contains of line with either directives or resource records
  - Directives are control entries that affect the rest of the zone file
  - If entries end with „.“, they are fully qualified, otherwise related to origin
- FQDN
  - Fully qualified domain name
  - Specifies the **exact** location in the hierachy tree
  - Topmost layer is the root zone (.)

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS zone file

```
$ORIGIN example.com.  ; designates the start of this zone file in the namespace
$TTL 3600  ; default expiration time (in seconds)

example.com     IN SOA ns.example.com. username.example.com. ( 2020091025 7200 3600 1209600 3600 )
example.com.    IN NS ns  ; ns.example.com is a nameserver for example.com
example.com.    IN NS ns.somewhere.example. ; ns.somewhere.example is a backup nameserver for example.com

example.com.    IN MX 10 mail.example.com.  ; mail.example.com is the mailserver for example.com
@               IN MX 20 mail2.example.com. ; equivalent to above line, "@" represents zone origin

example.com.    IN A 192.0.2.1  ; IPv4 address for example.com
                IN AAAA 2001:db8:10::1  ; IPv6 address for example.com

ns              IN A 192.0.2.2  ; IPv4 address for ns.example.com
                IN AAAA 2001:db8:10::2  ; IPv6 address for ns.example.com

www             IN CNAME example.com.  ; www.example.com is an alias for example.com

wwwtest         IN CNAME www  ; wwwtest.example.com is another alias for www.example.com
mail            IN A 192.0.2.3  ; IPv4 address for mail.example.com
mail2           IN A 192.0.2.4  ; IPv4 address for mail2.example.com
file            IN AAAA 2001:db8:10::3 ; IPv6 address for file.example.com
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

DNS zone file

Fully qualified

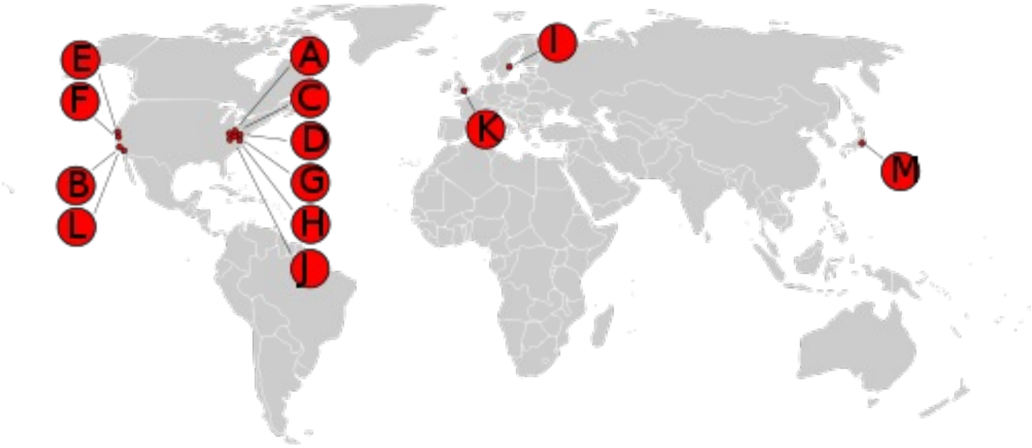
```
$ORIGIN example.com.  ; designates the start of this zone file in the namespace
$TTL 3600  ; default expiration time (in seconds)

example.com     IN SOA ns.example.com. username.example.com. ( 2020091025 7200 3600 1209600 3600 )
example.com.    IN NS ns  ; ns.example.com is a nameserver for example.com
example.com.    IN NS ns.somewhere.example. ; ns.somewhere.example is a backup nameserver for example.com

example.com.    IN MX 10 mail.example.com.  ; mail.example.com is the mailserver for example.com
@               IN MX 20 mail2.example.com. ; equivalent to above line, "@" represents zone origin

example.com.    IN A 192.0.2.1  ; IPv4 address for example.com
                IN AAAA 2001:db8:10::1  ; IPv6 address for example.com

ns              IN A 192.0.2.2  ; IPv4 address for ns.example.com
                IN AAAA 2001:db8:10::2  ; IPv6 address for ns.example.com

www             IN CNAME example.com.  ; www.example.com is an alias for example.com

wwwtest         IN CNAME www  ; wwwtest.example.com is another alias for www.example.com
mail            IN A 192.0.2.3  ; IPv4 address for mail.example.com
mail2           IN A 192.0.2.4  ; IPv4 address for mail2.example.com
file            IN AAAA 2001:db8:10::3 ; IPv6 address for file.example.com
```

Relative to origin example.com. => www.example.com.

Root Server

Fachhochschule Dortmund
University of Applied Sciences and Arts

- Name server for the root zone of the DNS of the internet (.)
- Answers requests related to the TLD (returns a list of authoritative servers)
- 13 logical root servers in the internet
  - a.root-server.net
  - b.root-server.net
  - …
  - m.root-server.net



Root Server II

- One small zonefile
  - Contains names and IP addresses of authoritative DNS servers for each TLD
  - Small (1.64MB as of 09 Jun 2023)
  - Changes infrequently (every couple of days)

- Q: How do I get to www.ub.fernuni-hagen.de?
- A: “I don’t exactly know, but you should ask the .de TLD server at:
  - a.nic.de, 194.0.0.53
  - s.de.net, 195.243.137.26
  - and others

Fachhochschule Dortmund
University of Applied Sciences and Arts

Root zone

Online available at https://www.internic.net/domain/root.zone
Entries for TLD .de

```
de.                 172800  IN      NS      a.nic.de.
de.                 172800  IN      NS      f.nic.de.
de.                 172800  IN      NS      l.de.net.
de.                 172800  IN      NS      n.de.net.
de.                 172800  IN      NS      s.de.net.
de.                 172800  IN      NS      z.nic.de.
<snip>
a.nic.de.     172800  IN      A       194.0.0.53
a.nic.de.     172800  IN      AAAA    2001:678:2:0:0:0:0:53
f.nic.de.     172800  IN      A       81.91.164.5
f.nic.de.     172800  IN      AAAA    2a02:568:0:2:0:0:0:53
<snip>
```

DNS management

- The ICANN (Internet Corporation for assigned Names and Numbers) and the IANA (Internet Assigned Numbers Authority) manage these top level domains and assign subdomains to organizations, in Germany e.B. the DENIC (http://www.denic.de).
- The organization may then independently manage the labels below its subdomain. Either the next label is a computer name or the organization divides its domain into further subdomains.
- The FH Dortmund has the subdomain fh-dortmund.de. This could be further subdivided, e.B. into dvz.fh-dortmund.de, verw.fh-dortmund.de, informatik.fh-dortmund.de, etc. These subdomains are then managed by the institutions mentioned in the label.

TLD

The following table lists some of the top level domains:

Geographical Top-level domains		Top-level domains	
at	Austria	com	commercial companies
ch	Switzerland	edu	educational institutions
de	Germany	gov	government institutions
es	Spain	int	international organisations
fr	France	mil	military organisations
au	Australia	net	network operator
ca	Canada	org	non-commercial organisations

In 2000, ICANN decided to introduce seven new top level domains:

.aero aviation industry , .biz company , .coop cooperative organizations, .info without restriction , .museum museums , .name individuals , .pro lawyers, tax consultants, doctors

In the meantime, others have been added: e.B. . jobs, .mobi, .travel in 2005

In 2023, .zip was added (plus .dad, .phd, .prof, .esq, .foo, .mov, and .nexus.)

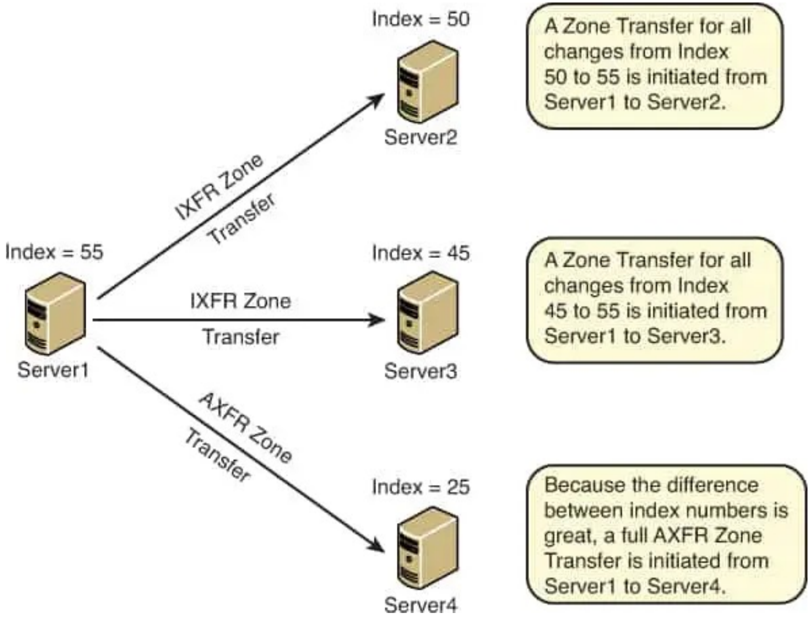
DNS servers and zones

- The entity that has **authority** over a zone is responsible for maintaining the DNS servers corresponding to that zone (authoritative servers)
- There are 2 types of **authoritative servers**:
  primary (only one) and **secondary**.
- A primary server for one zone can be a slave for a different zone simultaneously
- The secondaries update their information automatically, consulting the primary (SOA registry) and doing a "**zone transfer**".
  - RR type AXFR (*Asynchronous Full Transfer Zone* or *Asynchronous Xfer Full Range*
  - Zone transfer ensures replication of DNS DB
  - Transfer is done with Port 53/TCP (reliability is necessary)
  - The secondary server(s) updates the RRs from the primary server
  - Zone transfer allows distribution of the DNS
  - Improved version is IXFR (Incremental zone transfer), which only transfers changed RRs

Fachhochschule Dortmund
University of Applied Sciences and Arts

AFXR and IXFR example

- All 4 servers are equal
  - Every server can response to requests
  - Zone file contains (at least) 4 NS entries
- Changes are done on server 1 (master)
- Server 2-4 can be distributed all over the world
- If one server breaks, name resolution still possible



Inserting records into DNS

- Example: new startup "Network Utopia"
- register name networkuptopia.com at *DNS* *registrar* (e.g., Network Solutions)
  - provide names, IP addresses of authoritative name server (primary and secondary)
  - registrar inserts two RRs into com TLD server:

```
(networkutopia.com, dns1.networkutopia.com, NS)
(dns1.networkutopia.com, 212.212.212.1, A)
```

- Install DNS server and configure correct zone file
  - create authoritative server type A record for www.networkuptopia.com
  - type MX record for networkutopia.com
  - Insert A, AAAA and further RR into the zone file
  - Keep entries up-to-date
  - Increase serial no (in SOA RR) with each change

Recursive and caching only

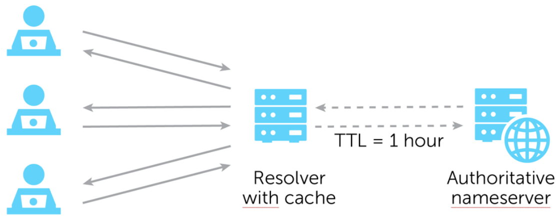
- On the client side, the resolution is usually mediated by a special server, called a **recursive and caching-only nameserver**, in which no information about any zones is configured.
- Caching-only server does entire name resolution process for the requesting client
- Server can use set TTL of a RR

```
mail            1h      IN A 192.0.2.3
a.nic.de.       172800  IN      A       194.0.0.53
```

TTL of 48h

TTL of 60 min

DNS cache time to live (TTL)



Fachhochschule Dortmund
University of Applied Sciences and Arts

Name resolving process

- In order for hosts to participate in the Domain Name System, they must use a Domain Name Resolver.
- This resolver software makes requests to the local domain name server to translate domain names into Internet addresses. Either the necessary information is available in the local name server or it must consult other name servers in order to provide the desired implementation.

```
Local Host                        |  Foreign
                                  |
   +---------+               +----------+  |  +--------+
   |         | user queries  |          |queries  |  |        |
   |  User   |-------------->|          |---------|->|Foreign |
   | Program |               | Resolver |         |  |  Name  |
   |         |<--------------|          |<--------|--| Server |
   |         | user responses|          |responses|  |        |
   +---------+               +----------+         |  +--------+
                               |     A            |
               cache additions |     | references |
                               V     |            |
                             +----------+         |
                             |  cache   |         |
                             +----------+         |
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

PTR

- Used for Reverse DNS
  - Mapping of an ip-address to an hostname
- The domain **in-addr.arpa (for IPv4)** is used for PTR
  - **ip6.arpa** used for IPv6

- Example:
  - 42.2.168.192.in-addr.arpa. 3600 IN PTR lab.example.local.
  - 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa. 3600 IN PTR ipv6.example.local.

lab.example.local has IP 192.168.2.42

ipv6.example.local has IP 2001:db8::1

Fachhochschule Dortmund
University of Applied Sciences and Arts

Extensions

- DNS is send in plain text over the channel
- Eavesdropping is possible
  - Everyone with access to the communication is able to read the requests and responses
- “New” extensions prevent the plain text transmission
  - DoH, DNS over HTTPS
  - DoT (RFC 7858), DNS over TLS
  - DoQ (RFC 9250), DNS over QUIC

DNS and Wireshark

- Full support for DNS protocol
- Relevant filters:
  - dns.a, dns.aaaa: Filter for dns responses
  - dns.resp.name == FQDN: Filter for DNS responses with FQDN in it
  - dns.qry.name == FQDN: Filter for DNS requests for FQDN
  - dns.cname: Filter CNAMEs

Fachhochschule Dortmund
University of Applied Sciences and Arts

Tools

- dig
- nslookup
- host

```
└─$ dig -t A www.fh-dortmund.de

; <<>> DiG 9.10.6 <<>> -t A www.fh-dortmund.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17578
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.fh-dortmund.de.            IN      A

;; ANSWER SECTION:
www.fh-dortmund.de.     10      IN      A       193.25.16.26

;; Query time: 74 msec
;; SERVER: fe80::d487:62:9df4:e3dd%15#53(fe80::d487:62:9df4:e3dd%15)
;; WHEN: Thu Jun 08 22:29:28 CEST 2023
;; MSG SIZE  rcvd: 63
```

```
└─$ host www.fh-dortmund.de
www.fh-dortmund.de has address 193.25.16.26
```

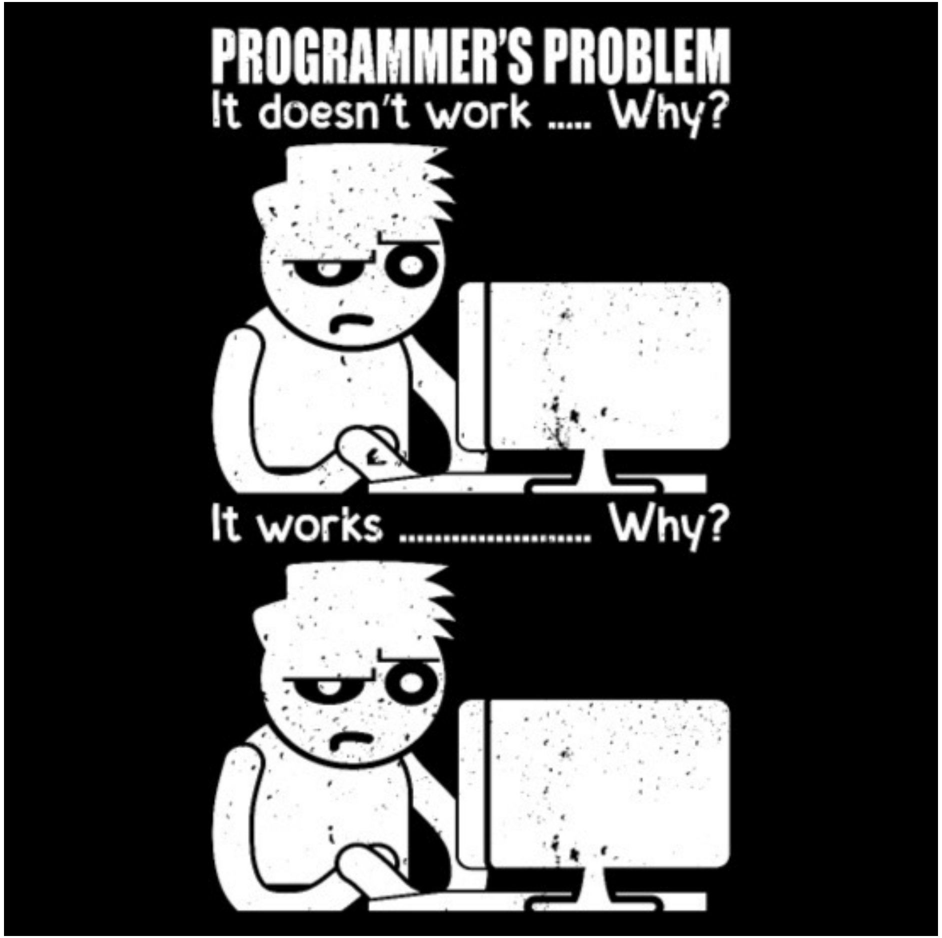
```
└─$ nslookup www.fh-dortmund.de
Server:         fe80::d487:62:9df4:e3dd%15
Address:        fe80::d487:62:9df4:e3dd%15#53

Non-authoritative answer:
Name:   www.fh-dortmund.de
Address: 193.25.16.26
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

Application Layer – HTTP and HTTPS

Prof. Dr. Daniel Spiekermann
Faculty of Computer Science

Fachhochschule
Dortmund
University of Applied Sciences and Arts

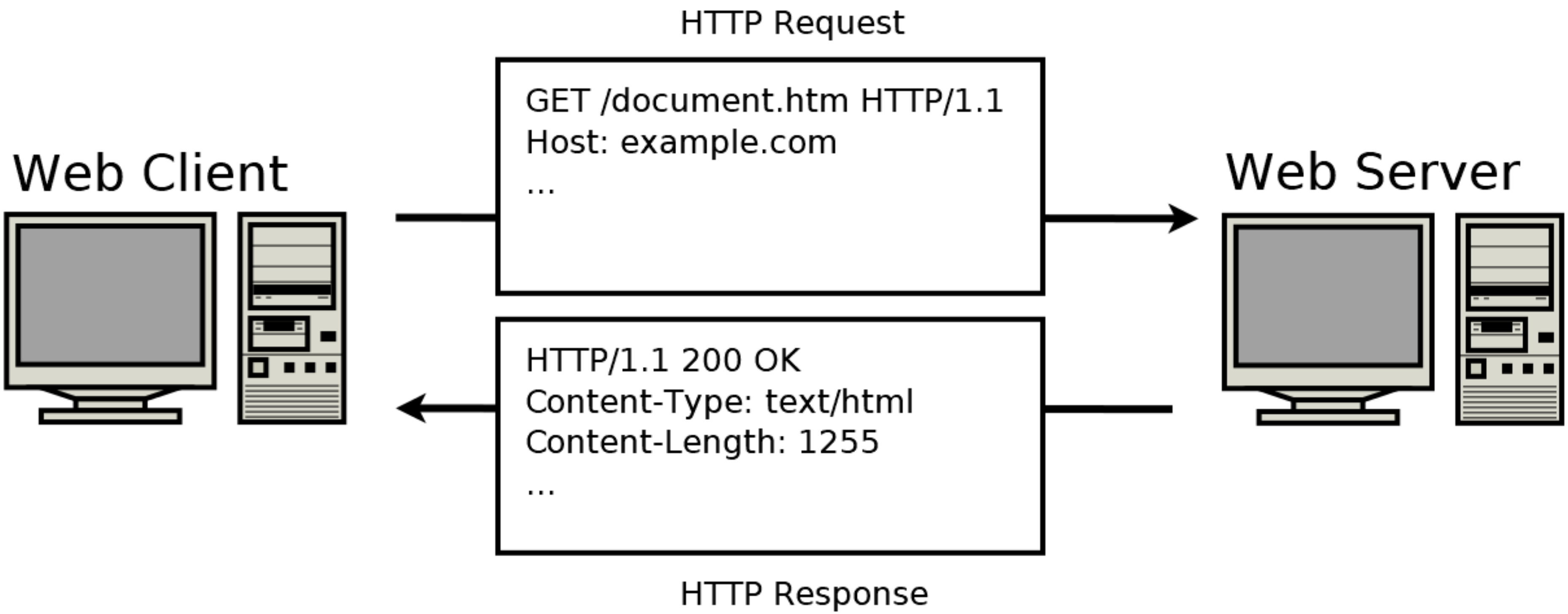


HTTP

- Hyper Text Transfer Protocol
- Developed by Tim Berners-Lee in 1989 at CERN
- Request/Response Protocol
- Stateless
- Uses Uniform Resource Locators (URL) for identifying and locating files
- Different versions
  - HTTP/1.0 1996 obsoletes
  - HTTP/1.1 1997
  - HTTP/2 2015
  - HTTP/3 2022
- Nowadays basement for various on-top protocols

Fachhochschule Dortmund
University of Applied Sciences and Arts

Client Server Communication



Communication

- Client tries to connect to a server establishing a connection via TCP 3-Way HS
- Server listening on that port accepts the connection and then waits for a client's request message (Successful 3-way HS)
- Client sends its **request** to the server
- Upon receiving the request, the server sends back an HTTP **response message** (header plus a body if it is required).
    - The body of this message is typically the requested resource, although an error message or other information may also be returned.
- At any time client or server can close the connection (TCP FIN or RST)
- Closing a connection is usually advertised in advance by using one or more HTTP headers in the last request/response message sent to server or client.

Fachhochschule Dortmund
University of Applied Sciences and Arts

Requests

HTTP defines a set of **request methods** to indicate the desired action

- **GET:** Requests a representation of the specified resource. Requests using GET should only retrieve data.
- HEAD: Asks for a response identical to a GET request, but without the response body
- **POST:** Submits an entity to the specified resource, often causing a change in state or side effects on the server.
- PUT: Replaces all current representations of the target resource with the request payload.
- DELETE: Deletes the specified resource.
- CONNECT: Establishes a tunnel to the server identified by the target resource.
- OPTIONS: Describes the communication options for the target resource.
- TRACE: Performs a message loop-back test along the path to the target resource.
- PATCH Applies partial modifications to a resource.

Header fields

- List of strings send from client or server
- Typically invisible to end-user
  - Only browser and server need the data
- Define various parameters
  - Encoding
  - User-agent
  - Caching and Aging
  - Compression
  - Content-statistics (length, type, hash,…)

Requests format

- Requests contains
- Request line: Request method
  - Requested URL
  - Protocol version
- At least 1 header field:
  - Field name: Value

```
(GET /index.html HTTP1.1)
```

```
Connection: keep-alive
Cache-Control: max-age=0
Accept-Encoding: gzip, deflate, br
```

Response format

- Response message from server contains
- Status line:
  - Protocol version
  - Status code
  - Optional phrase

```
HTTP/1.1 200 OK
```

- At least 1 response header field:
  - Field name: Value

```
Server: Apache/2.4.57 (Unix)
Last-Modified: Mon, 11 Jun 2007 18:53:14 GMT
Content-Type: text/html
```

Status Codes (RFC 9110)

Status codes are issued by a server in response to a client's request

- 1xx: Informational response
  request was received process is continued
- 2xx: Successful
  Request was successfully received, understood, and accepted
- 3xx: Redirection
  further action needs to be taken in order to complete the request
- 4xx: Client error
  Request contains bad syntax or cannot be fulfilled
- 5xx: Server error
  Server failed to fulfil the request

Wireshark and Status Codes

Protocol	Length	Info
HTTP	481	HTTP/1.1 301 Moved Permanently
HTTP	1012	HTTP/1.1 404 Not Found (text/html)
HTTP	71	HTTP/1.1 404 Not Found (text/html)
HTTP	1012	HTTP/1.1 404 Not Found (text/html)
HTTP	985	HTTP/1.1 404 Not Found (text/html)
HTTP	379	HTTP/1.1 302 Found
HTTP	972	HTTP/1.1 201 Created (text/javascript)
HTTP	299	HTTP/1.1 408 Request Time-out (text/html)
HTTP	591	HTTP/1.1 301 Moved Permanently (text/html)

Fachhochschule Dortmund
University of Applied Sciences and Arts

Communication

User:

10.0.0.1/index.html

Communication

User:

10.0.0.1/index.html

Browser:

```
GET /index.html HTTP 1.1

Host: 10.0.0.1
User-Agent: Mozilla/5.0
Accept:
text/html,application/xhtml+xml,applicatio
n/xml;q=0.9,image/avif,image/webp,*/*;q=0.
8
Accept-Language: en-GB,en;q=0.5 Accept-
Encoding: gzip, deflate, br Connection:
keep-alive
```

Communication

User:

10.0.0.1/index.html

Browser:

```
GET /index.html HTTP 1.1

Host: 10.0.0.1
User-Agent: Mozilla/5.0
Accept:
text/html,application/xhtml+xml,applicatio
n/xml;q=0.9,image/avif,image/webp,*/*;q=0.
8
Accept-Language: en-GB,en;q=0.5 Accept-
Encoding: gzip, deflate, br Connection:
keep-alive
```

Server:

```
HTTP/1.1 200
OK Date: Sat, 03 Jun 2023
14:14:23 GMT
Content-Type: text/html;
charset=UTF-8
Content-Length: 155
Last-Modified: Wed, 08 Jan
2003 23:11:55 GMT
Server: Apache/1.3.3.7
(Unix) (Red-Hat/Linux)
ETag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytes
Connection: close
<html> <head> <title>An
Example Page</title>
</head> <body> <p>It
works!</p> </body> </html>
```

Communication

User:

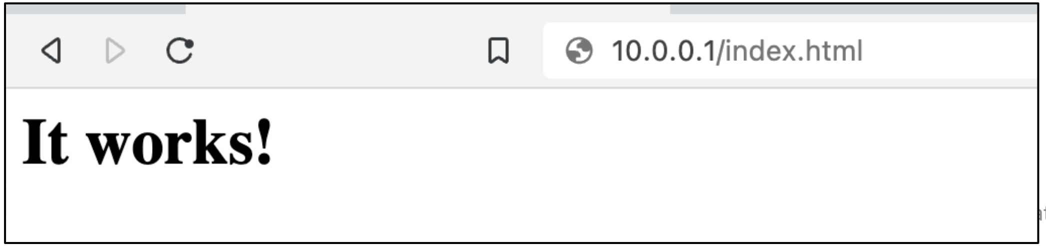
10.0.0.1/index.html

Browser:

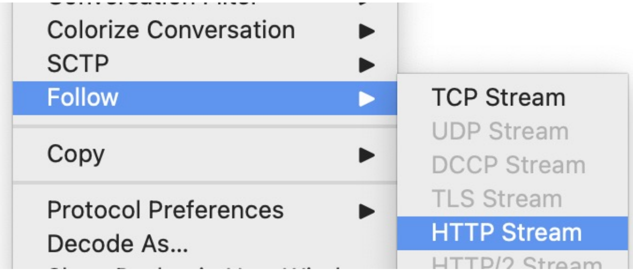
```
GET /index.html HTTP 1.1

Host: 10.0.0.1
User-Agent: Mozilla/5.0
Accept:
text/html,application/xhtml+xml,applicatio
n/xml;q=0.9,image/avif,image/webp,*/*;q=0.
8
Accept-Language: en-GB,en;q=0.5 Accept-
Encoding: gzip, deflate, br Connection:
keep-alive
```

Server:

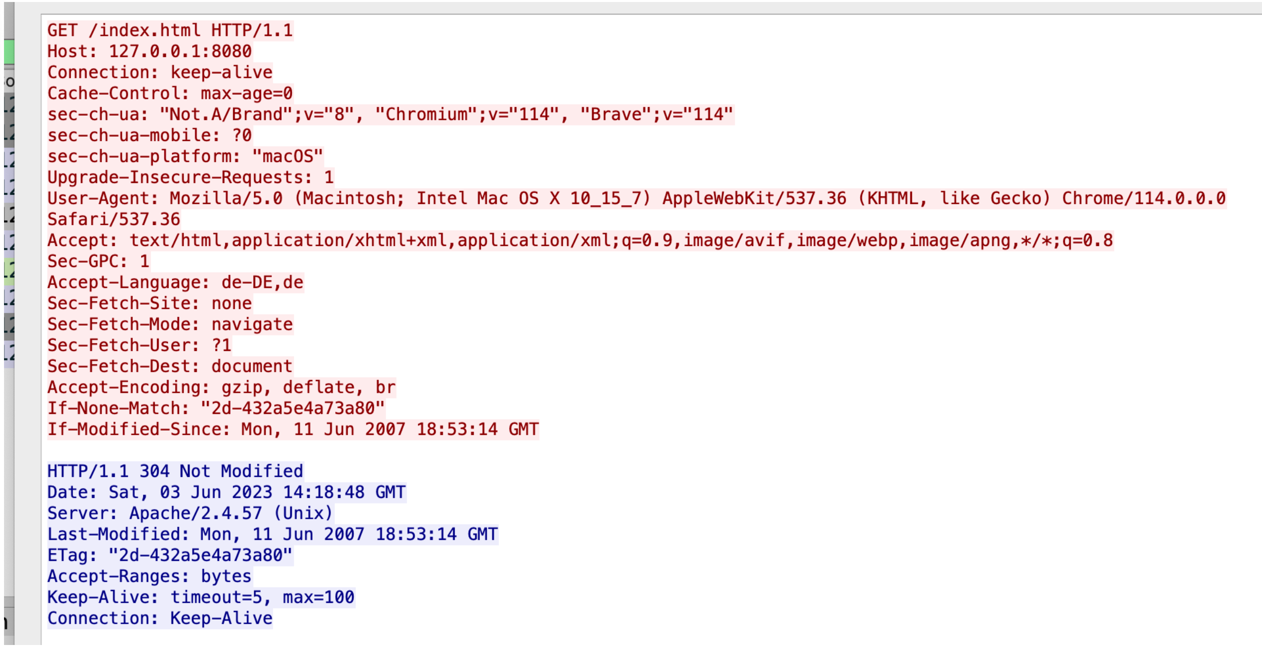
```
HTTP/1.1 200
OK Date: Sat, 03 Jun 2023
14:14:23 GMT
Content-Type: text/html;
charset=UTF-8
Content-Length: 155
Last-Modified: Wed, 08 Jan
2003 23:11:55 GMT
Server: Apache/1.3.3.7
(Unix) (Red-Hat/Linux)
ETag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytes
Connection: close
<html> <head> <title>An
Example Page</title>
</head> <body> <p>It
works!</p> </body> </html>
```



Wireshark – Analyze HTTP



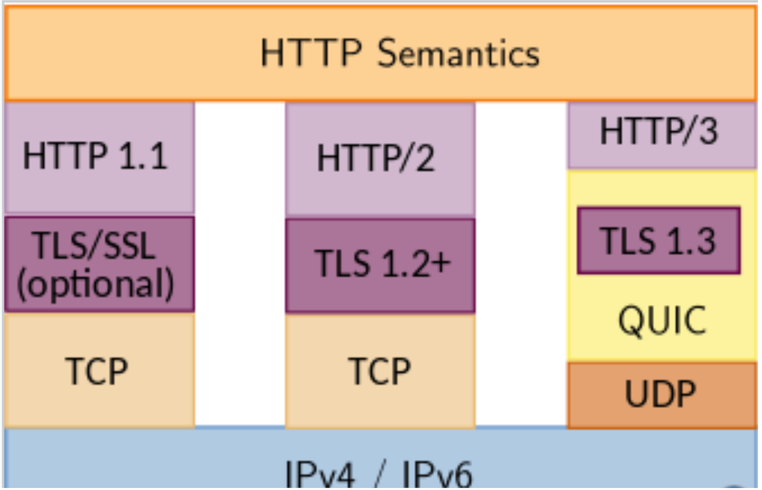
- Reconstructs the full communication process
  - Client request
  - Server response
- All header fields displayed



Improvement of HTTP

- HTTP/2 improves HTTP/1.1
  - Provides one or more bidirectional streams (prevents HOLB, Head of Line Blocking)
  - Less latency
  - Compression of header

- Current version is 3 (2022) as a revision of HTTP/2 (2015)
  - Uses QUIC+UDP instead of TCP



HTTPs

- HTTP transfers data in clear text
- Everyone with access to the connection can read the transferred data
- Eavesdropping of the connection gives access to all information
- Encryption does not prevent eavesdropping, but the clear text access

```
GET /index.html HTTP/1.1
Host: 127.0.0.1:8080
Connection: keep-alive
Cache-Control: max-age=0
sec-ch-ua: "Not.A/Brand";v="8", "Chromium";v="114", "Brave";v="114"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "macOS"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.0.0
Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8
Sec-GPC: 1
Accept-Language: de-DE,de
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br
If-None-Match: "2d-432a5e4a73a80"
If-Modified-Since: Mon, 11 Jun 2007 18:53:14 GMT

HTTP/1.1 304 Not Modified
Date: Sat, 03 Jun 2023 14:18:48 GMT
Server: Apache/2.4.57 (Unix)
Last-Modified: Mon, 11 Jun 2007 18:53:14 GMT
ETag: "2d-432a5e4a73a80"
Accept-Ranges: bytes
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

HTTPs encryption

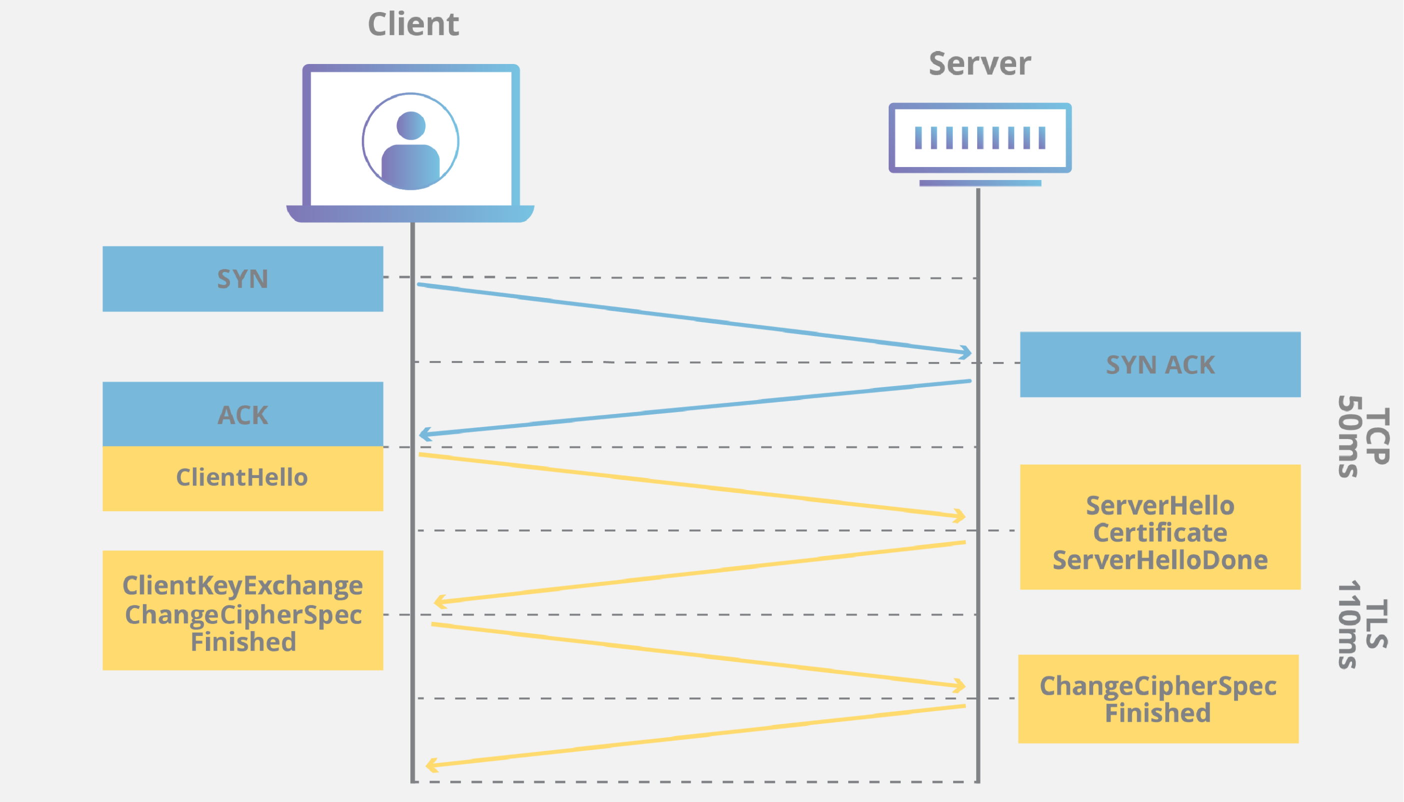
- HTTP over TLS
- HTTPS piggybacks HTTP entirely
- TLS (Transport Layer Security) after 3-way HS

Source	Destination	Protocol	Length	Info
192.168.179.203	193.99.144.85	TCP	78	59770 → 443 [SYN]
193.99.144.85	192.168.179.203	TCP	74	443 → 59770 [SYN,
192.168.179.203	193.99.144.85	TCP	66	59770 → 443 [ACK]
192.168.179.203	193.99.144.85	TLSv1.3	583	Client Hello
193.99.144.85	192.168.179.203	TCP	66	443 → 59770 [ACK]
193.99.144.85	192.168.179.203	TLSv1.3	1506	Server Hello, Cha
193.99.144.85	192.168.179.203	TCP	1506	443 → 59770 [ACK]
192.168.179.203	193.99.144.85	TCP	66	59770 → 443 [ACK]
192.168.179.203	193.99.144.85	TCP	66	59770 → 443 [ACK]
193.99.144.85	192.168.179.203	TLSv1.3	1506	Application Data
193.99.144.85	192.168.179.203	TLSv1.3	304	Application Data

HTTPS process

- Connection establishment from the client C to the server S over TCP 3-Way HS
- C sends cipher list (Client Hello)
- S authenticates with its certificate and chosen certificate (Server Hello, Certificate)
- C checks the cert and might send its own cert (Client Key Exchange, Client Certficate)
- C sends a random number (encrypted with pubkey of the server) or C and S use DH (Change Cipher Spec)
- All messages are encrypted with the calculated symmetric key

Communication



Client Hello

- The message includes the client’s TLS version, a random value, and a list of supported cipher suites.

```
⌄ TLSv1.3 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 512
  ⌄ Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 508
      Version: TLS 1.2 (0x0303)
      Random: 16e86faf3ea23de2200413913c679b9d22c2f14308cea2ee1cc1a74a4214c482
      Session ID Length: 32
      Session ID: 784d1aae9ea109c0e5b510791772f986e77ca521b34b68271a96a0c4828818
      Cipher Suites Length: 42
    › Cipher Suites (21 suites)
      Compression Methods Length: 1
    › Compression Methods (1 method)
      Extensions Length: 393
    › Extension: Reserved (GREASE) (len=0)
    › Extension: server_name (len=23)
    › Extension: extended_master_secret (len=0)
    › Extension: renegotiation_info (len=1)
```

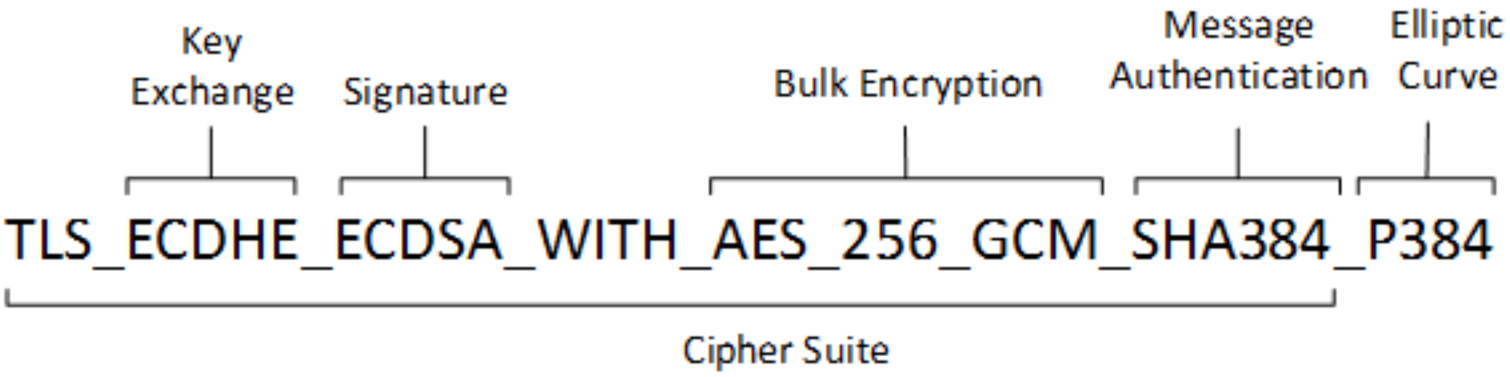
Server Hello

- The server will respond by sending a Server Hello message which includes the TLS version, random value, and chosen cipher suite.

```
⌄ Transport Layer Security
  ⌄ TLSv1.3 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 122
    ⌄ Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 118
        Version: TLS 1.2 (0x0303)
        Random: 78896013fa3021086e15296018f1d0d381c76220cd44a707c13e0502e2a3e819
        Session ID Length: 32
        Session ID: 784d1aae9ea109c0e5b510791772f986e77ca521b34b68271a96a0c4828818
        Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
        Compression Method: null (0)
        Extensions Length: 46
      › Extension: supported_versions (len=2)
      › Extension: key_share (len=36)
        [JA3S Fullstring: 771,4865,43-51]
```

Fachhochschule Dortmund
University of Applied Sciences and Arts

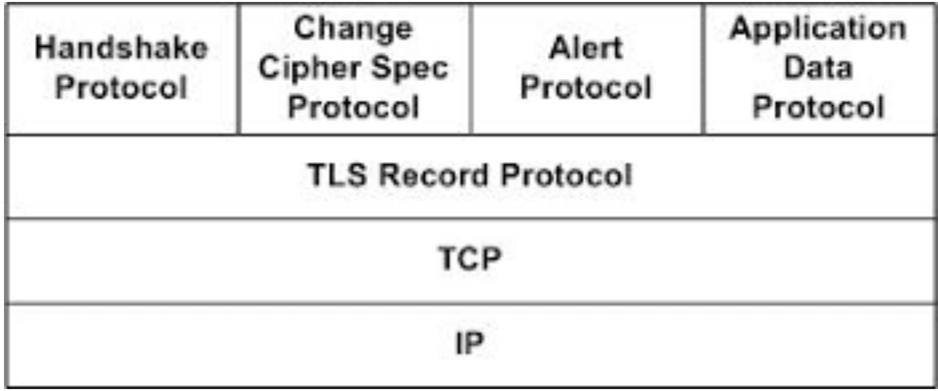
Cipher Suite



Client Key Exchange

- After the client receives the server certificate, it will perform a series of verification steps to validate the certificate.
- If the result is positive, then the client will continue to the Client Key exchange.
- Per master Secret is encrpyted with certificate of the server
- Session secret is calculated on both sides with PMS, client and server random
- Both sides have now the same session secret (sym. enc)
- Client sends ClientHelloDone, Server sends ServerHelloDone

TLS Record Layer

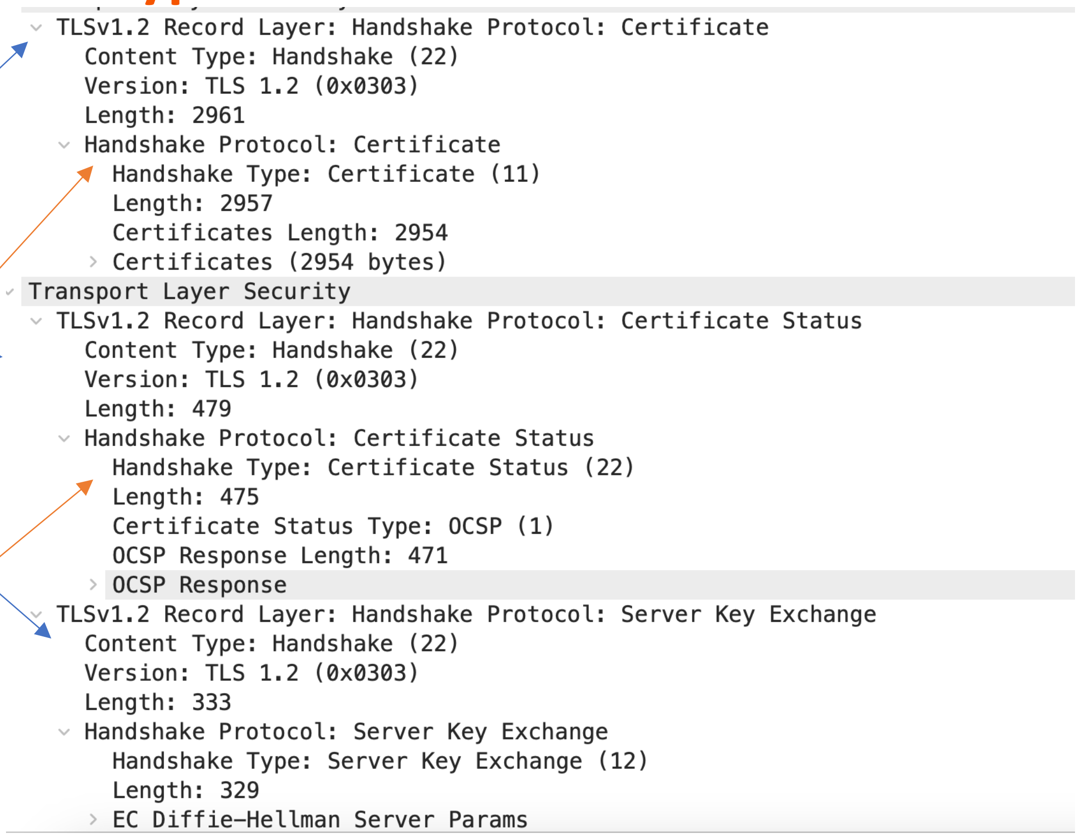


- Securing the TLS session
- Data is processed
  - Fragmentation (blocks of max 2^{14} Byte)
  - Compression (no default)
  - MAC (Message Authentication Code) for integrity
  - Encryption
- Format of TLS Record:
- Content Type | Protokollversion Major (1 Byte) | Protokollversion Minor (1 Byte) | Length (1 Short or 2 Byte)

Fachhochschule Dortmund
University of Applied Sciences and Arts

HTTPS Content Types

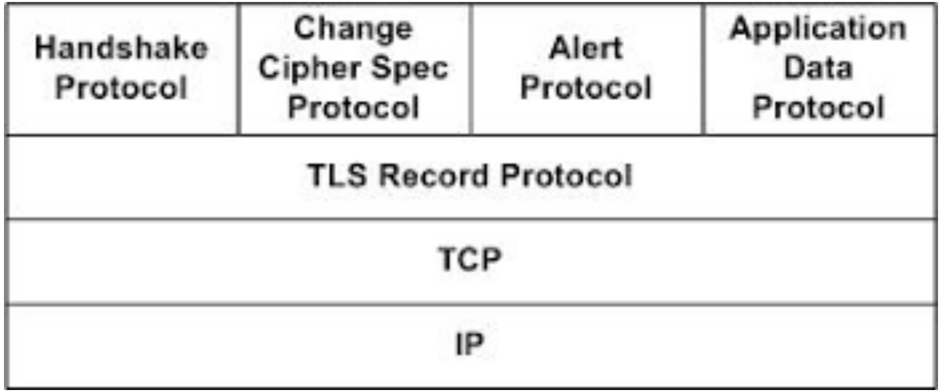
- Content Types
- 20 – Change Cipher Spec
- 21 – Alert
- 22 – Handshake
  - Handshake Type:
    - 1: Client Hello
    - 2: Server Hello
    - 4: New Session Ticket
    - 11: Certificate
    - 12: Serer Key Exchange
    - 14: Server Hello Done
    - 16: Client Key Exchange
    - 22: Certificate Status
- 23 – Application Data



Fachhochschule Dortmund
University of Applied Sciences and Arts

TLS Layer

- Handshake Protocol (Negotiation of session parameters)
- Alert Protocol (Management of alerts, warning or fatal. Fatal breaks connection)
- Change Cipher Spec Protocol (Changing to the negotiated cipher)
- Application Data Protocol (The transfer of the data over the TLS RR)



HTTPS and OSI

- Precise classification difficult
  - HTTPS is not a new protocol, it is HTTP with TLS
  - Layer 6 translates or formats data for the application layer based on the semantics or syntax the application accepts. This layer also handles the **encryption** and decryption that the application layer requires.
  - But TLS established a new session, which is part on L5

So TLS is sometimes assigned to L6, sometimes to L5, but sometimes to L7

OSI is not straightforward defined, it is a reference

Fachhochschule Dortmund
University of Applied Sciences and Arts

Additional slides

Prof. Dr. Daniel Spiekermann
Faculty of Computer Science