1 The Internet

To answer the respective questions, the following sources were used:

- *Historical information [Str23]*
- *Working of the Internet [CH23; Str21a; TP23; Dej20; Str21b]*

1 1 How does the Internet work?

The term Internet refers to the global system of interconnected computer networks using the internet protocol suite (TCP/IP) to communicate between networks and devices. Roughly speaking it serves to transport data to users requesting that data.

As a prototypical example, consider an user as client trying to access a website with the uniform resource locator (URL) https://www.google.com which has the domain name google.com. It consists of the top-level domain (TLD) com, the second-level domain (SLD) google and the subdomain www. To request the website from the server hosting it, the internet protocol (IP) address corresponding to the domain name, both of which are unique identifiers, has to be inferred. For that purpose the browser requests the IP address from the domain name system (DNS) server of the internet service provider (ISP) of the user. Physically, the request is send as electric pulses along an ethernet cable or as a frequency modulated electromagnetic wave through Wi-Fi to a switch which passes it on to a modem. Alternatively, switch and modem are combined in a gateway. The switch or gateway bridge the gap between local area network (LAN) and wide area network (WAN) and connect it the to the Internet. To reach the DNS server or other servers, the request possibly has to pass several routers which are switches connecting different networks and direct the traffic between them. The job of a DNS server is to translate domain names into IP addresses. While domain names are easier to remember for humans, IP addresses are easier to handle for computers. However there is an entire hierarchy of DNS servers to distribute the load of requests by different clients. If the DNS server asked does not have the IP address requested, it passes the request further up the hierarchy until it either finds the IP address and returns it or reaches the root. From there, the request goes to a DNS server responsible for the TLD, then to the SLD and finally to the subdomain(s). Once the IP address is found, it is returned to the browser. As a next step, the browser requests the website from the server hosting it. The server assembles the website, but splits it up into packets of $1-1.5\,\mathrm{kB}$ size consisting of the data as payload, a header and a footer. While the header contains the IP addresses of the sender and receiver as well as information to reassemble the original website, the footer signifies the end of the packet and includes measures to check for errors that might have happened during the transport. The packets are sent separately, possibly along different routes towards the computer which made the request. This way the load within the network of routers can be balanced and outages can be evaded. Once the packets arrive at their destionation, they are reassembled into the website which is then displayed by the browser. To speed up further requests to display this website, the browser caches the IP address.

Hence the Internet is a network of networks. While the LAN may belong to an user or a company, it connects via a modem to the network of their ISP. The networks of different ISPs connect to each other at internet access points (IXPs). Main routes of large ISPs or connections between IXPs are called the Internet backbone. They offer a wide bandwidth (amount of data that can be sent during a specific unit of time, typically: 1 s) and span large distances (e.g. deep-sea cables between the US and Europe). The size of a network and the conditions to transport data in other networks determines its association to one of three tiers. Although they are connected, using other networks to transport data of their customers can be handled in two ways: If the networks connected are large and of similar size, they may have peering agreements where data can traverse the other network free of charge while the same holds for data coming from the other network (tier 1). Smaller networks may have partially peering agreements and purchase IP transit, i.e. traffic across the other network is allowed for a fee (tier 2). Small regional networks only have access by means of IP transit (tier 3).

1.2 Why do we have Internet access almost everywhere?

This question can be answered on a historical and a technical level.

Historically, the predecessor of the Internet called ARPANET was funded by the US military to optimally use computational resources and as a communication network. Various national postal, telegraph and telephone services established and operated their own networks. A commercial version of ARPANET called Telenet started in 1975 and served commercial and government interests. With advances in semiconductor technology and optical networking the network was expanded to the public and commercialized during the 1990s.

Technically, there is a plethora of ways to access the Internet. One can use a computer modem via telephone circuits or broadband via coaxial cables, fibre optics or copper wires. Wireless methods such as Wi-Fi, satellite and cellular phone technology (e.g. 3G, 4G) via cell phone towers form another possibility. For sparsely populated or low-income areas, balloons and ground stations to transmit radio waves can be used.

1.3 How do you use the Internet?

Collecting information (google, research articles on arXiv, APS), Entertainment (Youtube, Webcomics), Staying connected (Family, Friends), Learning (Online Courses), Shopping (amazon), Navigation (Google maps)

1.4 Is the Internet dangerous?

Making the distinction between the Internet as infrastructure and the world wide web as the information system using the Internet, the question is interpreted as referring to the world wide web. In recent years several incidents ranging from social media posts pressuring teenagers into eating disorder [Cra19], rumors spread through messenger services leading to lynchings in India [Sam20], the broadcast of a mass shooting via a social media website [BBC19] and the proliferation of hate-speech [Hat18] have shown that the world wide web can have devastating real-life consequences. The web enables humans to manipulate or harm other humans mentally by cyberbullying or people can radicalize themselves by consuming media displaying extreme viewpoints who then harm others physically. In this sense the web is similar to a car which can be abused to harm people and can lead to accidents, but also have its benefits.

1.5 What interest you about the Internet?

I am curious about the optimization algorithms used to determine the optimal route for packets since it is a type of traveling salesman problem.

1.6 Why was the Internet developed?

The predecessor of the Internet called ARPANET was an indirect product of the Sputnik crisis during the Cold War. As a reaction to the Sowjet Union launching the worlds first artificial satellite, Eisenhower founded the Advanced Research Projects Agency (ARPA) in 1958, which developed the network. The aim was to share resources between remote computers (“time-sharing”) to allow researchers access to supercomputers without having to travel to them. Starting in 1969, ARPANET first connected two, then several american universities and research institutions. It developed into a decentralized communication network within the US, and started to spread to Europe in 1973 via Norway and England. The connection to England formed the first internetwork, whose short form internet was first used in a 1974 article on “A Protocol for Packet Network Intercommunication”. This term was fully adopted in 1982, when the internet protocol suite (TCP/IP) was standardized which allowed for a worldwide growth of interconnected networks.

2 Network engineering

To answer the respective questions, the following sources were used:

- *First network: [Fou23b]*
-
- *Purpose of network engineering: [Kom23c]*
- *Network components: [Fou23c; Kom23b; Cis23]*
- *Client-server communication: [Fou23a; Kom23a]*

2 1 Which were the first networks and what was their purpose?

Based on the context, it is assumed that this question refers to computer networks using packet switching, hence telephone networks (circuit switching) and telegraph networks (message switching) used for telecommunication are not considered.

The first two packet switched computer networks which went online in 1969 were the local area Mark I packet-switched network at the National Physical Laboratory (NPL) in the UK and the wide area ARPANET in the US connecting three universities in California and one in Utah. While the NPL network aimed to prove the feasibility of package switching and provide access to a common computer from a growing number of terminals, the ARPANET intended to share resources between remote computers ("time-sharing") to allow researchers access to supercomputers without having to travel to them. The Merit network in the US, which went online in 1971, intended to connect the mainframe computers of three universities in Michigan. In contrast the CYCLADES computer network was designed to improve internetworking. Public data networks such as the commercial Telenet in the US (1975), DATAPAC in Canada (1976) and TRANSPAC in France (1978) offered services to commerical and public users.

2 2 What are networks necessary for?

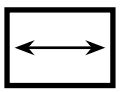
The purpose of a network is to connect devices (e.g. phones, computers, servers, routers) which allows them to exchange information.

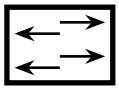
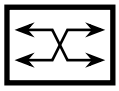
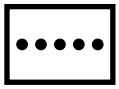
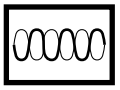
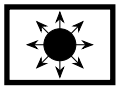
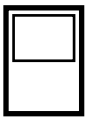
2 3 What is the purpose of network engineering?

Network engineers are responsible for designing, implementing and maintaining the internal network infrastructure, IT support and system administration in a company. Hence the purpose of network engineering at large is to fulfill these tasks.

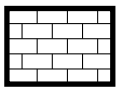
2 4 Which network components are there?

- Links: Patch cables, installation cables, sockets, connectors, patch panels, patch cabinet. Passive components of a network. The symbol for links are straight lines.
- Hub: Connects ethernet devices within a network and makes them act as a single network segment. This is accomplished by passing the input it receives at any of its input/output (I/O) ports to all other ports. In the OSI model, hubs operate on the physical layer (layer 1). Hubs are largely obsolete. The symbol for a hub is:

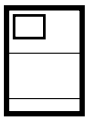


- Bridge: Creates a single, combined network from multiple communication networks or network segments. While routers allow independent communication between multiple networks while they remain separate, bridges connect two separate networks as if they were a single network. In the OSI model, bridges operate on the data link layer (layer 2). The symbol for a bridge is: 
- Switch: A switch connects devices within a network (often LAN) and forwards frames to and from those devices by means of packet switching, i.e. routing and transferring data by means of addressed packets such that the transmission channel is only occupied during transport. It is a multiport bridge that uses MAC addresses to forward data only to the intended receiver. In the OSI model it thus operates on the data link layer (layer 2). However there also exist multilayer switches which have IP addresses, incorporate routing functionality and forward data at the network layer (layer 3). Besides unmanaged switches which have no configuration interface or options, there are managed and enterprise switches which have IP addresses and can be configured and remotely administered. The symbols for a switch are:  
- Router: Forwards packets between computer networks. Along their journey, packets travel from router to router until they reach their destination. Upon arrival of a packet, a router determines from the network address information in the packet header the final destination. Using its routing table or routing policy, the router determines the available routes, selects the most suitable route considering different criteria, activates the physical connection to other networks, adjusts the packets by means of segmentation and directs the packet to the next network on its journey. In the OSI model, routers operate on the network layer (layer 3). The symbol for a router is: 
- Modem: Transforms data between digital format and a format suitable for an analog transmission medium such as telephone or radio for sending and receiving it. When sending, it modulates one or more carrier wave signals to encode digital information and demodulates it when it receives data. The symbol for a modem is: 
- (Wireless) access point: Enables Wi-Fi devices to connect to a wired network. This is accomplished by acting like a Hub, i.e. it passes the input received to all connected devices (→prefer Ethernet cable). It can have a wired connection to a router or be part of a router. In the OSI model, (wireless) access points operate on the data link layer (layer 2). The symbol for a (wireless) access point is: 
- Repeater: Receives signals and retransmits it to cover longer distances or pass an obstruction. It can be an amplifier or a combination of receiver and transmitter. In the OSI model, repeaters operate on the physical layer (layer 1). The symbol for a repeater is: 
- Gateway: Allows flow of data from one network to another. In contrast to switches and routers, gateways communicate using more than one protocol to connect multiple networks. In the OSI model, gateways can operate on any of the seven layers. The symbol for a gateway is: 
- Network interface card/controller (NIC): Connects a computer to a computer network either by cable (OSI: physical layer, layer 1, ethernet protocol) or wirelessly (OSI: data link layer, layer 2, Wi-Fi protocol) This enables communication between computers within the same LAN as well as within a large-scale network with routable protocols (e.g. IP). The computers are identified by means of their MAC addresses. The symbol for a NIC is: 
- Firewall: Network security system which monitors and controls incoming and outgoing network traffic

based on predetermined security rules. It serves as a barrier between a trusted and an untrusted network (e.g. Internet) and operates on the network or transport layer (layer 3 or 4). The symbol for a firewall is:



- Media converter: Converts signals between different types of cables.
- Server: Provides services (functionalities, e.g. sharing data, performing computations) to other devices called clients. This architecture is called client-server model. A single server can serve multiple clients and a single client can use multiple servers. Client processes can run locally or on servers. There exists a plethora of different types servers: webserver (hosts websites), mail server (sends and receives emails), DNS server (translates domain names into IP addresses), file server (hosts storage), database server (hosts structured or unstructured databanks), groupware server (supports work in groups), proxy server (acts as stand-in for clients to save bandwidth and improve access speed by caching data), streaming/-media server (host audio- and video files), print server (connects printer to the network). The symbol for a server is:



- Load balancer: Distributes the load of many customer requests onto multiple web server instead of having a single one to deal with the demand. The load balancer gets the public domain while the webserver get enumerated subdomains like www1, www2, etc. or country abbreviations.

2 5 Which end devices are there?

- Printer
- Phones, Tablets
- Computers: Workstations, Laptops
- NAS

2 6 What is client-server communication?

The client-server model refers to a distributed application which divides tasks between clients and servers. While clients request resources or services, servers provide them. This model can be applied to separate machines which communicate over a network as well as to the communication between processes within a single machine. The communication between clients and servers consists of messages in the request-response/reply messaging pattern, where the client sends a request and the server responds. Communication protocols provide the common language and rules of the communication which the client and server have to follow. In the OSI model they operate in the application layer (layer 7).

Within the client-server model, the server plays the central role in the network (server-based networks). Servers are classified by the service they provide, communication between clients runs via servers and they must be permanently available and running. Clients initiate the communication with the server and have to understand the response based on the protocol (content, formatting of data for the requested service) while they are oblivious to the inner workings of the server. The communication can be formalized by implementing an application programming interface (API) which serves as an abstraction layer (way to hide the inner workings) for accessing the service. Depending on the application requiring the service functions, a computer can be the client, the server or both. On the software level, client processes can communicate with server processes. In case one wants to explicitely address computers, one talks about client- and server-hosts. There are different ways how the communication proceeds:

- synchronous: client only sends next request after previous request has been answered
- pipeline: client sends several requests in a sequence and expects the corresponding responses
- asynchronous: client sends multiple requests, but the responses arrive possibly in a different order and at different times

A problem associated with the client-server model is the possibility to overload the server by excessive requests during denial of service (DoS) attacks. This can be combatted by limiting the availability of the server for clients

via a scheduling system. In case sensitive infromation are passed between client and server, the requests can be encrypted. Examples for protocols are http, https (world wide web) and smtp, imap, pop (email).

2 7 Problem 1: Home network

2 7 1 List of components

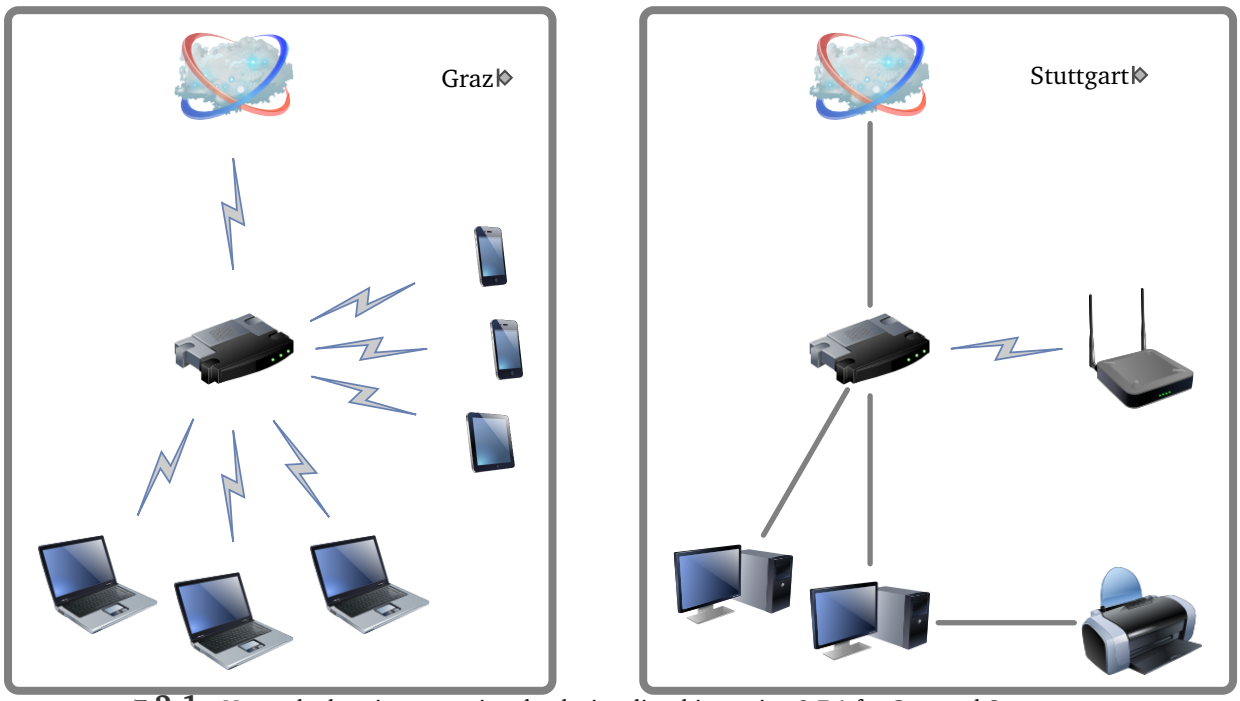
Network components:

- Router (Graz)
- Router (Stuttgart)
- Switch (Stuttgart)
- Repeater (Stuttgart)
- Ethernet cable with RJ45 (Graz, Stuttgart)

End devices:

- Laptop ×3 (Graz)
- Smartphone ×2 (Graz)
- Tablet (Graz)
- Desktop Computer ×2 (Stuttgart)
- Printer (Stuttgart)

2 7 2 Network plan



2.1 – Network plans incorporating the devices listed in section 2.7.1 for Graz and Stuttgart.

TODO: Add switch

3 OSI reference model

To answer the respective questions, the following sources were used:

3 1 What was the reason to develop layer models?

Networking is a complicated subject since it involves many hardware and software elements. To better understand such a complex system, it is helpful to break it down into pieces and analyze their function and how they interact. Therefore it is reasonable to divide the overall set of functions into modular components, with each of which having their own role. Concerning the interaction between components, interfaces between them have to be defined.

In Networking, functions are divided into layers which contain hardware and/or software elements. Each layer is responsible to perform a type of task and interact with the layers above and below it. Lower layers are responsible for concrete tasks such as hardware signaling and low-level communication, while higher layers combine these services to implement more abstract functions.

3 2 What is the purpose of the OSI model?

The purpose of the Open systems interconnection (OSI) model introduced in 1984 was originally to serve as the foundation to establish a widely-adopted suite of protocols to be used by international internetworks. Today this purpose if fulfilled by TCP/IP, so the OSI model is used to explain networking in general terms. This goes beyond educational purposes and covers the interaction between components of other protocol suites and hardware devices.

3 3 What is the purpose of each of the seven layers?

7. Application: While this layer does not include applications themselves, it enables them to communicate over the network. It consists of application services which provide network applications with a plethora of functionalities. Additionally, application advertisement takes place in this layer, where devices on the network offering a service (e.g. a printer) inform other devices about that service. The protocol data unit (PDU) of this layer is the (application) message. Examples for network applications and protocols are:
    - Web surfing: HyperText transfer protocol (HTTP, HTTPS) to send and receive websites
    - File transfer: File transfer protocol (FTP) to transfer files on the Internet
    - Email: Post office protocol version 3 (POP3) use by Email clients to retrieve messages from remote servers, Simple mail transferring protocol (SMTP) to email messages on the Internet, Internet message access protocol (IMAP) to email messages on the Internet
    - Chat: Internet relay chat protocol (IRC) to chat over the Internet
    - Remote access: Teletype network (TELNET) to access remote systems in plaintext, Secure shell (SSH) to access remote systems using a secure channel and an encrypted format
    - IP addresses: Domain name system (DNS) to translate IP addresses into domain names and vice versa, Dynamical host configuration protocol (DHCP) to automatically assign IP addresses to computer in a network
6. Presentation: This layer is responsible for translation, compression and encryption/decryption of data:
    - Translation: Conversion of data to machine-understandable format (e.g. characters, numbers, ASCII format to binary, EBCDIC format)

- Compression: Reducing the number of bits reduces the amount of space required to store data and allows faster transmission of data (e.g. real-time audio or video streaming). Such data compression can be lossy (e.g. raster graphics such as JPEG, PNG) or lossless (e.g. vector graphics such as EPS, PDF).
- Encryption/Decryption: Maintain the integrity of data and enhance the security of sensitive data by having the sender encrypt and the receiver decrypt the data. The secure socket layer (SSL) protocol ist used.

5. Session: This layer helps in setting up, maintaining and terminating sessions or connections to send and receive data. It consists of tools which are provided to higher layer protocols via sets of command called application program interfaces (APIs). The main tasks handled in this layer are authentication, authorization and session management:

   - Authentication: Before a session/connection with a server is established, the client has to verify (e.g. provide username and password) to the server who they are.
   - Authorization: The server checks during a session, whether the client has permissions to access the requested files. If the client does not have the permissions, the server informs the client of their lack of permissions.
   - Session management: The session layer keeps track of files that are downloaded during a session. For example when requesting a website the texts and images of which are stored on a server, the web browser opens a separate session to download these files from the server. The packets received, the files they belong to (e.g. text or video) and where they move (e.g. web browser) are tracked by the session layer.

   One example of an API is the network basic input/output system (NETBIOS) which allows applications on different computers to communicate. Examples for protocols are the layer 2 tunneling protocol (L2TP), remote transport control protocol (RTCP) and the H.245 protocol.

4. Transport: This layer controls the reliability of the communication. It takes the (application) messages received from the application layer as service data units (SDUs), breaks them into parts and wraps the them with a header into (TCP) segments/(UDP) datagrams as PDUs.
   In general, this layer is responsible for:

   - Process-level addressing: To provide multiple different processes and network services with access to network layer protocols simultaneously, they receive addresses. In the transmission control protocol (TCP) and the user datagram protocol (UDP) these addresses are called port numbers.
   - Multiplexing and Demultiplexing: Using addresses allows to multiplex, i.e. combine data from multiple different applications into a single stream of data to be sent. At the receiver, the incoming stream of data is demultiplexed, i.e. disentangled and the (TCP) segments/(UDP) datagrams are directed to the recipient applications.
   - Segmentation, packaging and reassembly: (Application) messages are segmented, i.e. split into smaller pieces called (TCP) segments or (UDP) datagrams to fit them into (IP) packets of the network layer. At the receiver the (application) messages are reassembled.
     While the only noticeable contents of the header of (UDP) datagrams are the source and destination port numbers as well as a checksum, the header of (TCP) segments have more relevant content besides the port numbers and a checksum:
     - Sequence number: For normal transmissions, the sequence number of the first byte of data in the segment. In a connection request with the SYN control bit, the initial sequence number of the source TCP. Used to reassemble (TCP) segment at the destination.
     - Acknowledgement number: Used in combination with the ACK control bit and contains the sequence number the source expects the destination to send next.
     - Control Bits:
       - Urgent Bit (URG): Invoke priority data transfer feature for segment.
       - Acknowledgement Bit (ACK): Segment carries acknowledgement, the acknowledgement number field is valied and it contains the sequence number expected next from the destination of the segment.
       - Push Bit (PSH): TCP push feature, requesting that the data in the segment be immediately pushed to the application in the receiving device.
       - Reset Bit (RST): Sender has encountered a problem and wants to reset the connection.
       - Synchronize Bit (SYN): Request to synchronize sequence numbers and establish a connection. The sequence number field contains the initial sequence number of the sender of the segment.

    - Finish Bit (FIN): Sender of segment requests to close connection.
  - Window: Number of octets of data the sender of the segment is willing to accept from the receiver at one time, which corresponds to the size of the buffer allocated to accept data for the connection. More formal, it is the current receive window of the sending device and the send window of the receiving device.

- Connection establishment, management and termination: Connection-oriented protocols as TCP are responsible for a series of communications required to establish, maintain and terminate a connection. To establish a connection, TCP uses a three-way handshake between the sender S and the receiver R:
  - S →R: SYN
  - R →S: SYN-ACK
  - S →R: ACK

  To terminate a connection, TCP uses a four-way handshake between the sender S and the receiver R:
  - S →R: FIN
  - R →S: ACK
  - R →S: FIN
  - S →R: ACK

  Note that UDP is connectionless.
- Acknowledgements and retransmissions: Connection-oriented protocols as TCP guarantee reliable data using automatic repeat requests (ARQs). Once data is sent, a timer is started. If the data is received and successfully checked for errors, the recipient sends back an acknowledgement to indicate a successful transmission. If no acknowledgement comes back before the timer expires, the data is resend. Hence erroneous and lost data are accounted for.
- Flow control: Restrict the amount of data the sender can send, to prevent overwhelming the receiver. Since the receiving devices have limited processing speed and memory to store incoming data, exceeding those limits means loss of incoming data. To avoid this, the receiver asks the sender to slow down before the limits are met.

Examples for protocols used are the reliable, connection-oriented TCP and the faster, connectionless UDP. Devices operating at this layer are firewalls since they examine (TCP) segments/(UDP) datagrams to make decide on transmission.

3. Network: This layer serves to transport data between devices on different networks. It takes the (TCP) segments/(UDP) datagrams received from the transport layer as SDUs and wraps them with a header into (IP) packets as PDUs. The header contains besides the sender and receiver IP addresses (without subnet mask) information about fragmentation, the time to live (TTL) and the transport or network layer protocol carried. In general, this layer is responsible for:

   - Logical Addressing: Each device communicating over a network has a logical/layer 3 address. In case of the Internet, they are IP addresses. These addresses are independent of particular hardware and are unique across an entire internetwork. The network layer uses IP addresses to ensure that data intended for a specific device on a different network reaches its destination.
   - Routing: Moving data across a series of interconnected networks. This entails for routers to handle incoming packets from various sources, determine their final destination and figure out where to send them to get them (closer) to their destination.

     In case devices are located on the same network, routing is not required and an ARP request provides the destination MAC address for a given destination IP address. If devices are located on different networks, routing is required since an ARP request is restricted to a network. First, the sending device determines its default gateways MAC address using an ARP request. Then the device connects via the default gateway through routers to the destination network, where the local router infers the destination MAC address via an ARP request. This way, the sending device can address the data with the correct IP and MAC address to get it delivered to its destination. Note that while the source and destination IP addresses do not change during routing, the source and destination MAC addresses do change.

     Path determination, i.e. the determination of the best possible among many paths from sender to receiver, is an aspect of routing where protocols come into play.
   - Packet encapsulation: Encapsulation of (TCP) segments/(UDP) datagrams into (IP) packets with a header (total length in bytes, identification, flags, fragment offset, TTL, protocol, sender and receiver IP addresses). The mentioned, relevant fields in the header have the following meaning:

- Identification: Identify fragments of the same (IP) packet.
- Flags: If the DF (Don't fragment) flag is set to 1, the (IP) packet must not be fragmented. In case an (IP) packet arrives at a router which is too large for a physical medium but which has the DF flag set to 1, it is discarded. If the MF (More fragments) field is set to 0, it is the last fragment of a message, while 1 means there are more fragments to come. In case no fragmentation takes place, the MF field is 0.
- Fragment offset: Identify the location of a fragment within an (IP) packet.
- TTL: Maximum number of hops an (IP) packet may perform before the next router discards it.
- Protocol: Identifies the transport or network layer protocol carried in the (IP) packet. Examples are 0x01 for ICMP, 0x06 for TCP and 0x11 for UDP.

▸ Fragmentation and reassembly: Since the (IP) packets are passed down to the data link layer, where the different technologies have limits on the length of an (Ethernet) frame, the network layer has to perform fragmentation, i.e. splitting up the (IP) packets. At the final receiver, the (IP) packets have to be reassembled. The length limit on the data link layer is called maximum transmission unit (MTU) which amounts for Ethernet to 1500 byte.

Note that the MTU is essentially determined by the physical medium used to transport the data, e.g. a twisted pair cable may imply Ethernet with an MTU of 1500 byte, while the absence of a wired connection implies IEEE 802.11 with an MTU of 2304 byte. However the physical medium can change from source to intermediate routers to destination. It can thus be necessary for routers to further fragment (IP) packets, while reassembly at routers is not performed. The minimum MTU amounts to 576 byte. During MTU path discovery, the path with an optimal MTU during routing is determined by using ICMP messages with decreasing size and the DF flag set to true.

▸ Error handling and diagnostics: Exchange information between devices which are logically connected or trying to route traffic, about the status of the hosts on the network or the devices themselves.

Examples for protocols used are the Internet protocol (IP), Internet control message protocol (ICMP), IP network address translation (NAT) and IPSec. Note that IP is a connectionless protocol with no guarantee of delivery and no error checking. These are responsibilities of the transport layer. A selection of routing protocols are the routing information protocol (RIP), border gateway protocol (BGP), open shortest path first (OSPF) protocol and intermediate system to intermediate system (IS-IS) protocol. Devices operating at this layer are routers and firewalls since they examine (IP) packets to make transport decisions.

2. Data link: This layer serves to transport data between devices within the same network. It takes the (IP) packets received from the network layer as SDUs and wraps them with a header and trailer into (Ethernet) frames as PDUs. While the header contains the sender and receiver media access control (MAC) addresses, the trailer contains information to detect errors. The data link layer is split into the logical link control (LLC) sublayer and the MAC sublayer. In general, this layer is responsible for:

▸ LLC: Functions required for the establishment and control of logical links between local devices on a network. This entails flow control and error control.

Flow control refers to restricting the amount of data the sender can send, to prevent overwhelming the receiver. Since the receiving devices have limited processing speed and memory to store incoming data, exceeding those limits means loss of incoming data. To avoid this, the receiver asks the sender to slow down before the limits are met.

Error control refers to error detection and retransmission. Error detection is accomplished by performing a cyclic redundancy check (CRC). These are simple to implement, easy to analyze mathematically and very good to detect common errors caused by noise in transmission channels. Essentially a mathematical operation is performed on the SDU as bitstring and the resulting 4 byte are put into the trailer of the (Ethernet) frame. Upon receiving the (Ethernet) frame, the calculation is repeated and the result compared with the data in the trailer. Different values imply an error. Besides retransmitting erroneous data, lost data also has to be transmitted anew. For this purpose, automatic repeat requests (ARQs) are used. Once an (Ethernet) frame is received and checked for errors, the receiver sends an acknowledgement to the sender. Since lost or damaged (Ethernet) frames do not trigger an acknowledgement, they are automatically sent again.

Note that most services of the LLC sublayer are already provided by the transport layer. Hence these services are usually bypassed.

▸ MAC: Procedures used by devices to control access to the network medium. Since many networks use a shared medium (e.g. single network cable) it is necessary to have rules for managing the medium to avoid conflicts. While the Ethernet technology uses carrier sense multiple access with collision detection (CSMA/CD), the Token Ring technology uses token passing.

As an example, CSMA/CD has all computers listening to the cable before sending data through the network. If the cable is clear, the data is transmitted and if the cable is occupied, the computer waits until it is clear. If two computers send their data at the same time, a collision occurs. In this case, the involved computers stop transmission and wait random amounts of time before trying to retransmit. In this method it is normal for collisions to occur. Since collision and retransmission cause only a very small delay, the transmission speed of the network is normally not affected.

- Data framing: Encapsulation of (IP) packets into (Ethernet) frames with header (sender and receiver MAC addresses) and trailer (CRC).
- Physical addressing: Each device on a network has a unique physical/hardware/MAC address embedded in NICs by their manufacturer. It is a 12-digit/6 byte/48 bit string of hexadecimal numbers represented as six groups with two digits each, separated by hyphens, colons or without separator. The data link layer uses MAC addresses to ensure that data intended for a specific machine on the network reaches its destination.

The data link layer is formed by the software embedded in the NIC. It defines the technologies and standards implemented on the physical layer. These encompass hardware and encoding aspects of LAN and wireless LAN. Examples for technologies and protocols used are Ethernet, Token Ring, IEEE 802.11, the serial link interface protocol (SLIP) and the point-to-point protocol (PPP). Devices operating at this layer are switches, (wireless) access points, NICs and bridges since they examine (Ethernet) frames to make transport decisions.

1. Physical: This layer takes the (Ethernet) frames received from the data link layer as a binary sequence of bits, and converts it into signals transmitted over local media. These signals can be electrical signals transported over copper cables, light signals transported over optical fibre optical cables or radio signals transported through the air. On this layer, the PDU is the bit/byte. In general, this layer is responsible for:

   - Definition of hardware specifications: Details of operation of cables, connectors, wireless radio transceivers, NICs and other hardware.
   - Encoding and signaling: Encoding and signaling functions that transform data from bits on a device into signals sent over the network.
   - Data transmission and reception: Transmission and reception of data sent wired or wireless.
   - Topology and physical network design: Hardware related network design issues, e.g. LAN, WAN topology.

   The main protocols used are the Ethernet (twisted pair cable, coaxial cable, fibre optical cable), FastEthernet and GigabitEthernet (twisted pair cable, fibre optical cable) protocols. Devices operating at this layer are repeaters, hubs and transceivers (e.g. NICs).

Note that once the signals arrive at the physical layer of the receiver, they are converted into bits. In the data link layer, the bits are decapsulated into (Ethernet) frames, which are decapsulated in the network layer into (IP) packets and finally decapsulated in the transport layer into (TCP) segments/(UDP) datagrams. In case of TCP, the segments are assembled into (application) messages, while UDP forwards the datagrams to the protocol using its service.

3.4 What are bits/bytes, frames and packets?

Associate each term with a layer

Bits/bytes, (Ethernet) frames, (IP) packets and (TCP) segments/(UDP) datagrams are protocol data units (PDUs) of the physical, data link, network and transport layers. This means they are the units of data with which the protocols of the respective layers operate. A more precise description of the PDUs

- Bits/bytes: Payload consisting of an (Ethernet) frame considered as binary sequence.
- (Ethernet) frames: Header with MAC addresses of sender and receiver, trailer with checksum for CRC and payload consisting of an (IP) packet for each (Ethernet) frame.
- (IP) packets: Header with IP addresses of sender and receiver, information about fragmentation, TTL, transport or network layer protocol carried and payload consisting of a (TCP) segment/(UDP) datagram or a fragment thereof for each (IP) packet.
- (UDP) datagrams: Header with port numbers of sender and receiver, checksum and payload consisting of a small (application) message.

- (TCP) segments: Header with port numbers of sender and receiver, checksum, sequence number, acknowledgement number, control bits window size and payload consisting of a piece of an (application) message.

3 5 What is an IP address?

An internet protocol (IP) or logical address is a 32 bit (IPv4) or 128 bit (IPv6) long binary number represented by four octets in the range $[0, 255]$ separated by periods or eight groups of four hexadecimal digits separated by colons. It is a unique identifier for the interface of a device to the network and allows routers to identify that device within a network. Within the OSI reference model the source and destination IP addresses are added in the network layer as a header to an (TCP) segment/(UDP) datagram to form an (IP) packet. When data is transmitted through several networks, the source and destination IP addresses remain unchanged.

Special IPv4 addresses are:

Network-ID	Host-ID	Class A Example	Class B Example	Class C Example	Meaning
Network-ID	Host-ID	77.91.215.5/8	154.3.99.6/16	227.82.157.160/24	Normal IP
Network-ID	All 0	77.0.0.0/8	154.3.0.0/16	227.82.157.0/24	Specified network
All 0	Host-ID	0.91.215.5/8	0.0.99.6/16	0.0.0.160/24	Specified host on this network
Network-ID	All 1	77.255.255.255	154.3.255.255	227.82.157.255	All hosts on specified network
0.0.0.0/8			0.0.0.0		Me
255.255.255.255			255.255.255.255		All hosts on network
10.0.0.0/8			10.0.0.0 – 10.255.255.255		Private use
172.16.0.0/12			172.16.0.0 – 172.31.255.255		Private use
192.168.0.0/16			192.168.0.0 – 192.168.255.255		Private use
127.0.0.0/8			127.0.0.0		Loopback

Note that subnets, the network address (first IP address) and the broadcast IP address (last IP address) cannot be assigned to a device.

Special IPv6 addresses are:

- Private/local-use addresses:
  - First nine bits: 1111.1110.1... →Hexadecimal: FE8...–FEF...
  - scope: local network →communication sent to local devices
  - Site-local addresses:
    - First ten bits: 1111.1110.11... →Hexadecimal: FEC...–FEF...
    - scope: site, organization
  - Link-local addresses:
    - First ten bits: 1111.1110.10... →Hexadecimal: FE8...–FEB...
    - scope: physical link/network
    - purpose: address configuration, address resolution, neighbor discovery
- Loopback:
  - 0:0:0:0:0:0:0:1 (short: ::1)
  - sends IP packets back to device for testing purposes
- Unspecified address:
  - 0:0:0:0:0:0:0:0 (short: :: or 0::0)
  - used as source IP by device requesting its IP address to be configured

3 6 What is a physical address?

A media access control (MAC) or physical address is a 12-digit/48-bit/6-byte hexadecimal number embedded by the manufacturer in the network interface controller/card (NIC) of a networking device. It is a world-wide

unique identifier every networking device possesses and allows switches to identify a device within a network. Within the OSI reference model the source and destination MAC addresses are added in the data link layer as a header to an (IP) packet to form an (Ethernet) frame. When data is transmitted through several networks, the source and destination MAC addresses will change.

Special MAC addresses are:

- Broadcast: FF-FF-FF-FF-FF-FF
- Multicast: [01-00-5e-00-00-00,01-00-5e-7f-ff-ff]
- Placeholder: 00-00-00-00-00-00

3.7 Problem 2: OSI reference model

The network components and end devices mentioned in problem 1 are assigned to the layers of the OSI model as follows:

Network components/end devices	ISO/OSI layer	Reason
Cable	1 (physical)	Cables transport bits on the physical layer
Router	3 (network)	Routers operate on the network layer
Repeater	1 (physical)	Repeaters extend the physical range of a network
Laptops	1-7	
Computer	1-7	
Printer	1-7	
Tablet	1-7	
Smartphone	1-7	

3.8 Problem 3: Network engineering: Basic terms

1. Q: What is an internet protocol (IP) address?
   A: An IP or logical address is a 32 bit (IPv4) or 128 bit (IPv6) long binary number represented by four octets in the range $[0, 255]$ separated by periods or eight groups of four hexadecimal digits separated by colons. It is a unique identifier for a computer in a network and allows routers to identify a device within a network. Within the OSI reference model the source and destination IP addresses are added in the network layer as a header to an (TCP) segment/(UDP) datagram to form an (IP) packet. When data is transmitted through several networks, the source and destination IP addresses remain unchanged.
2. Q: What is the purpose of the commands ping, tracert, nslookup, route print and arp? Are these the same commands for Linux? If no, are there alternatives?
   A: The commands purposes and their counterparts on Linux are:
   - ping: Verifies the IP-level connectivity to another TCP/IP computer by sending Internet Control Message Protocol (ICMP) echo Request messages and returning the echo Reply messages along with the round-trip times the messages took. It serves to troubleshoot connectivity, reachability and name resolution. On Linux the command is also ping.
   - tracert: Determines the path taken to a destination by sending ICMP echo Request/ICMPv6 messages to the destination with incrementally increasing time to live (TTL) field values. Each router along the path decrements the TTL by one before forwarding it. Once the TTL value has reached zero, the router returns an ICMP time Exceeded message to the source computer. The TTL value thus counts the maximum links. Routers which do not return time Exceeded messages show up as a row of asterisks in the output. Once the message reaches its destination, tracert acts like ping. On Linux the command is traceroute.
   - nslookup: Information to diagnose the domain name system (DNS) infrastructure. It returns the IP address for a given domain name without attempting to reach it and thus tests the connection to DNS servers. On Linux the command is also nslookup, but one can also use the command dig.
   - route print: Displays the contents of the IP routing table. The route command can be used to display and modify entries in the local IP routing table, i.e. add or remove routes, specify the subnet mask, specify hop addresses and change the metric of a route. On Linux the command ip

route show table all is used to display the IP routing table. To modify entries of the IP routing table, the command ip route with its options is used.

- arp: Displays and modifies entries in the address resolution protocol (ARP) cache. This cache contains for each NIC a table which stores the IP addresses and their resolved MAC addresses. On Linux the command is also arp.

3. Q: What is the Internet control message protocol (ICMP)?
   A: The ICMP is a protocol in TCP/IP used by networking devices (e.g. routers) to send error messages and operational information indicating success or failure when communicating with another IP address. Its messages are used for diagnostic or control purposes (e.g. ping, tracert) or generated in response to errors in IP operations. In the OSI model it operates on the network layer (layer 3).
4. Q: What is the address resolution protocol (ARP)?
   A: The ARP is a protocol used to obtain the MAC address (link layer) associated with a given IP address (network layer) without crossing over to other networks. In the OSI model it operates on the data link layer (layer 2).
5. Q: What does the command "ping bearingpoint.com -t" do?
   A: This command sends continuously echo Request messages to the IP address associated with the domain name bearingpoint.com until stopped.
6. Q: How can an ARP-table be shown on a computer?
   A: The command arp -a accomplishes this on Windows while arp or arp -a (condensed output) are used on Linux.
7. Q: What does the command "arp -a" do?
   A: This command displays the current ARP cache tables for all interfaces.
8. Q: Determine the IP and MAC address of your computer. What is the command on the command line?
   A: Command: ipconfig /all (Windows)
   IP address: 10.157.80.88 (WLAN adapter)
   MAC address: 34.2E.B7.0C-BE-F8
   IP address: 192.168.56.1 (VirtualBox)
   MAC address: 0A-00-27-00-00-20
   Command: ip link (Linux)
9. Q: Determine the IP address of "a1.net". What is the command on the command line?
   A: Command: ping a1.net
   IP address: 80.75.40.1
10. Q: Which route (IP addresses) do packets take on their way from your computer to www.google.com?
    A: Command: tracert www.google.com
    Output:

    1 2 ms 1 ms 2 ms 10.157.80.2
    2 3 ms 3 ms 3 ms 195.3.81.21
    3 2 ms 2 ms 3 ms 195.3.81.241
    4 3 ms 3 ms 2 ms 172.17.65.53
    5 6 ms 5 ms 6 ms lg14-9080.as8447.a1.net [195.3.64.105]
    6 6 ms 5 ms 6 ms lg2-9071.as8447.a1.net [195.3.64.14]
    7 9 ms 9 ms 9 ms lg59-9071.as8447.a1.net [80.120.167.46]
    8 9 ms 9 ms 9 ms 172.253.51.153
    9 33 ms 13 ms 17 ms 192.178.81.124
    10 * * * Request timed out.
    11 26 ms 26 ms 26 ms 142.251.234.208
    12 39 ms 40 ms 40 ms 142.251.236.99
    13 39 ms 39 ms 39 ms 172.253.71.80
    14 41 ms 40 ms 40 ms 209.85.255.187
    15 * * * Request timed out.
    16 * * * Request timed out.
    17 * * * Request timed out.
    18 * * * Request timed out.
    19 * * * Request timed out.
    20 * * * Request timed out.
    21 40 ms 39 ms 38 ms di-in-f105.1e100.net [74.125.193.105]

11. Q: How many routers does the packet have to pass until it reaches www.google.com?

A: Including the final response, 21 routers reply. Whether the final response comes from the destination or from a server before it enters a firmnet the internal workings of which are unclear, is not known.

12. Q: What is the loopback (or local host) IP address of a computer and what is its purpose? Is it the same for Windows and Linux? Or are there differences?
A: The loopback IP addresses cover the range 127.0.0.0 to 127.255.255.255 and are used for loopback functionality. IP packets sent by a host to an address in this range are not passed down to the data link layer for transmission, but looped back to the sender at the IP level. The purpose of the loopback range is to test the TCP/IP implementation on a host, more specifically to test the layers above and including the network layer since it avoids the data link and physical layer. One address often used for testing purposes is 127.0.0.1 which can thus be called the loopback (or local host) IP address. Note that the boundaries 127.0.0.0 and 127.255.255.255 are actually excluded since the former specifies the entire network and the latter is used for broadcasting to all hosts on the local network.
13. Q: What is a hop?
A: A hop occurs when a packet passes from one network segment to another, i.e. passes a router. The number of hops required from source to destination is a rough measure for the distance between hosts and its limit is called time to live (TTL).
14. Q: How can a network connection between two computers be tested?
A: By using the ping command to ping one computer from the other.
15. Q: Which cable is required to connect two computers?
A: An Ethernet cross-over cable is required. In such a cable both ends use different wiring standards (one side T568A, the other T568B) such that one side has the transmit and receive signals interchanged compared to the other. This is necessary to connect two devices of the same type (e.g. two computers, two switches).
16. Q: What is the purpose of a default gateway?
A: The default gateway identifier is the IP address of the router which provides the default routing functions for a particular device. To send an (IP) packet to a device outside its network, it sends the (IP) packet to the default gateway which takes care of the routing.
17. Q: What is the Internet Assigned Numbers Authority (IANA) and what is the Réseaux IP Européens Network Coordination Centre (RIPE NCC)?
A: The IANA oversees the global IP address allocation, autonomous system (AS) number allocation, root zone management in the domain name system (DNS), media types and other IP related symbols and Internet numbers. It delegates these Internet resources to regional internet registries (RIRs) such as RIPE NCC which is responsible for Europe, the Middle East and parts of Central Asia. RIRs further delegate these resources to national and local internet registries (NIRs, LIRs) most of which are ISPs, enterprises or academic institutions which assign them to their customers.
18. Q: What is a domain name system (DNS) and what is its purpose?
A: The purpose of a name system is to translate between the name given to a device by humans and its numerical address used by computers. Specifically the domain name system (DNS) fulfills three basic name system functions:

- Name space: DNS uses a hierarchical name space. Starting from a single root into which containers (domains) are placed, each container can contain either individual device names or more specific subcontainers.
- Name registration: DNS name registration is used to enter names into the DNS distributed database. The authorities handling this registration are also arranged hierarchically with the central authority determining the overall structure of the name space while lower authorities determine their parts of the name space.
- Name resolution: To resolve a name into an address, DNS provides a distributed client-server name resolution mechanism, with name resolvers as the clients and name servers as the servers.
  - Name resolvers: When a user references a name in a network application, the name is passed to the resolver which requests information from a name server. They employ caching to improve performance.
  - Name servers: Servers maintained by organizations having administrative control over part of the DNS name space. They contain resource records describing names, addresses and other characteristics of those parts of the name space. The servers are thus also arranged hierarchically analogous to the name space. Their job is to receive requests for name resolution and either respond with the data requested, or with the name of another name server which will lead to the requested information. They employ caching to improve performance.

19. Q: Why can a computer possess multiple MAC addresses?

↳ A: Since a MAC address is assigned to each NIC and a computer can possess multiple NICs, a computer can also possess multiple MAC addresses.

20. Q: How is a MAC address also called?
↳ A: A media access control (MAC) or physical address is a 12-digit/6-byte hexadecimal number embedded by the manufacturer in the network interface controller/card (NIC) of a networking device. It is a world-wide unique identifier every networking device possesses and allows switches to identify a device within a network. Within the OSI reference model the source and destination MAC addresses are added in the data link layer as a header to an (IP) packet to form an (Ethernet) frame. When data is transmitted through several networks, the source and destination MAC addresses will change.
21. Q: Why is it important that a computer possess an IP address?
↳ A: Without an IP address, a computer could not communicate with other devices in a computer network since it could not be identified.
22. Q: What is a switch?
↳ A: A switch connects devices within a network (often LAN) and forwards frames to and from those devices by means of packet switching, i.e. routing and transferring data by means of addressed packets such that the transmission channel is only occupied during transport. It is a multiport bridge that uses MAC addresses to forward data only to the intended receiver. In the OSI model it thus operates on the data link layer (layer 2). However there also exist multilayer switches which incorporate routing functionality and forward data at the network layer (layer 3). Besides unmanaged switches which have no configuration interface or options, there are managed and enterprise switches which can be configured and remotely administered.
23. Q: Why can a LAN not connect arbitrarily many computers?
↳ A: There are not arbitrarily many computers in a LAN possible, since there is only a finite number of MAC and IP addresses available in a local network by means of which the computers could be addressed. However a more practical issue is the network traffic which will impact the networks performance. For example switches send out periodical broadcast messages to determine the state of the network, which will lead to the network congesting.
24. Q: What is a router?
↳ A: A router forwards packets between computer networks. Along their journey, packets travel from router to router until they reach their destination. Upon arrival of a packet, a router determines from the network address information in the packet header the final destination. Using its routing table or routing policy, the router determines the available routes, selects the most suitable route considering different criteria, activates the physical connection to other networks, adjusts the packets by means of segmentation and directs the packet to the next network on its journey. In the OSI model, routers operate on the network layer (layer 3).
25. Q: Which public IP address are you traversing the Internet with?
↳ A: I am using the public IP address provided by my ISP.
26. Q: What is an IP network?
↳ A: An IP network is a group of computers which exchange messages using the internet protocol.
27. Q: What is network address translation (NAT) and what is its purpose?
↳ A: NAT translates RFC1918 addresses, i.e. private IP addresses, into public IP addresses. It allows to delay running out of IPv4 addresses by having large numbers of hosts with private IP addresses in the same network share a small number of public IP addresses. Furthermore, different networks can use the same private IP addresses for their hosts and thus preserve IP addresses. NAT also allows the distinction between private and public addresses which adds a layer of protection against cyber threats since hosts with private IPs cannot be reached from outside the network.

3.9 Problem 4: Packet tracer

3.9.1 Problem 4.1: Simple Peer-to-Peer Network

1. The two computers are connected with a copper cross-over cable. In such a cable both ends use different wiring standards (one side T568A, the other T568B) such that one side has the transmit and receive signals interchanged compared to the other. This is necessary to connect two devices of the same type (e.g. two computers, two switches).
2. Configuration of IP address:

Command: ipconfig 192.168.12.1 255.255.0.0 192.168.0.1 (IP address, subnet mask, default gateway)

Computer	IP	MAC
PC0	192.168.12.1	000C.CF26.6EE7
PC1	192.168.12.254	0001.C7E9.6DDB

For simplicity, the IP and MAC addresses will be abbreviated as IP{0,1} and MAC{0,1}

3. Connectivity tests: Sequential, Parallel
   - PC0: ping IP1, tracert IP1
   - PC1: ping IP0, tracert IP0
   - PC0 & PC1: ping IP1 & ping IP0, tracert IP1 & tracert IP0
4. IP, MAC addresses in packets:

Addresses	PC0→PC1	PC1→PC0
SRC	IP0, MAC0	IP1, MAC1
DST	IP1, MAC1	IP0, MAC0

Protocols: Internet Control Message Protocol (ICMP)
↳ Progress diagrams:

- PC0: ping IP1 (PC0↔PC1 for PC1: ping IP0)

- PC0: tracert IP1 (PC0↔PC1 for PC1: tracert IP0)

- PC0: ping IP1 & PC1: ping IP0

- PC0: tracert IP1 & PC1: tracert IP0

PC0: arp -d, ping IP1

- PC0: arp -d

↳ ARP Broadcast (DST ADDR: FFFF.FFFF.FFFF) by PC0 (TGT MAC: 0000.0000.0000, TGT IP: IP0) →no response by PC1 since IP1 ≠ TGT IP = IP0

- PC0: ping IP1

↳ PC0 tries to send ICMP message, but header has no Ethernetll part (no MAC addresses)
↳ ARP Broadcast by PC0 (TGT IP: IP1) →reponse by PC1 with MAC1 since IP1 = TGT IP
↳ PC0 performs ping (4 × 3 exchanged ICMP messages) addressed to MAC1

PC0: arp -d, PC1: ping IP0

- PC0: arp -d

↳ ARP Broadcast (DST ADDR: FFFF.FFFF.FFFF) by PC0 (TGT MAC: 0000.0000.0000, TGT IP: IP0) →no response by PC1 since IP1 ≠ TGT IP = IP0

- PC1: ping IP0

↳ PC0 receives ICMP message and tries to respond, but header has no Ethernetll part (no MAC addresses)
↳ ARP Broadcast by PC0 (TGT IP: IP1) →reponse by PC1 with MAC1 since IP1 = TGT IP
↳ PC0 replies to ping (4 × 3 exchanged ICMP messages) addressed to MAC1

Remarks: Upon restarting a network, each PC sends out an ARP Broadcast, but none is answered since each uses their own IP as TGT IP.

3.9.2 Problem 4.2: Simple Network with a Switch

1. The devices are connected with a copper straight-through cable. In such a cable both ends use the same wiring standards (both sides T568A or T568B). Such a cable is sufficient to connect two devices of different type (e.g. a computer to a switch), but to connect two devices of the same type, a cross-over cable is necessary.
2. Configuration of IP address:
   Command: ipconfig 192.168.14.1 255.255.0.0 192.168.0.1 (IP address, subnet mask, default gateway)

Computer	IP	MAC
PC0	192.168.14.1	0090.21DB.05A5
PC1	192.168.14.2	0060.2FC2.2E8C
PC2	192.168.14.254	0002.1713.01A6

3. PC0: ping IP1, ping IP2, tracert IP1, tracert IP2
   PC1: ping IP0, ping IP2, tracert IP0, tracert IP2
   PC2: ping IP0, ping IP1, tracert IP0, tracert IP1
4. PC0: pint IP2
   - PC0 tries to send ICMP message, but header has no Ethernetll part (no MAC addresses)
   - ARP Broadcast by PC0 (TGT IP: IP2) →switch ignores request since it is busy sending out STP messages
   - PC0 tries to resend ICMP message, but header has no Ethernetll part (no MAC addresses)
   - PC0 tries to send ICMP message, but header has no Ethernetll part (no MAC addresses)
   - ARP Broadcast by PC0 (TGT IP: IP2) →switch ignores request since it is busy sending out STP messages
   - PC0 tries to resend ICMP message, but header has no Ethernetll part (no MAC addresses)
   - PC0 tries to send ICMP message, but header has no Ethernetll part (no MAC addresses)
   - ARP Broadcast by PC0 (TGT IP: IP2) →switch ignores request since it is busy sending out STP messages and sends out DTP messages
   - PC0 tries to resend ICMP message, but header has no Ethernetll part (no MAC addresses)
   - PC0 tries to send ICMP message, but header has no Ethernetll part (no MAC addresses)
   - ARP Broadcast by PC0 (TGT IP: IP2) →ARP broadcast by switch (TGT IP: IP2) →response to ARP broadcast by PC2 (SRC MAC: MAC2, TGT MAC: MAC0, SRC IP: IP2, TGT IP: IP0) →response to ARP broadcast by switch (SRC MAC: MAC2, TGT MAC: MAC0, SRC IP: IP2, TGT IP: IP0)
   - PC0 performs ping (1 exchanged ICMP message) addressed to MAC2

   Remarks: The switch memorizes the SRC MAC addresses of incoming messages. All MAC addresses memorized by the switch can be shown within the CLI of the switch by entering

   ```
   Switch> enable
   Switch# show mac-adress-table
   ```

   Note that when checking for the precise timing when MAC0 is memorized by the switch, this seems to happen at a random ARP message. The memorization also does not seem to have an impact on when the broadcast is forwarded by the switch. It seems that the broadcast is only forwarded after the DTP messages have been sent out, although there were instances when the ARP broadcast was forwarded before any STP message could be sent out.

4 IP addresses

To answer the respective questions, the following sources were used:

- *[Koz17]*

4 1 Since when does IPv4 exist?

The Internet protocol version 4 (IPv4) was defined in the Request for Comments (RFC) 791 in 1981.

4 2 What are RFC1918 addresses and what are they used for?

An RFC1918 address is an IP address assigned by a private organization to an internal host within a private network. As such they are neither available, nor reachable from the Internet, which allows them to be used multiple times in different private networks isolated from each other. Hence only public IP addresses are unique within each internetwork. Technically this is accomplished by configuring routers to discard IP packets carrying RFC1918 addresses. Besides the original motivation to delay running out of IPv4 addresses, the distinction between private and public addresses adds a layer of protection against cyber threats. It also condemns hosts with RFC1918 addresses to the role of a client. Addresses in the RFC1918 range are 10.0.0.0/8, 172.16.0.0/12 and 192.168.0.0/16. A type of private addresses reserved for automatic private address allocation is 169.254.0.0/16 which are assigned if a DHCP server cannot be reached and the device would be without IP address.

4 3 What is the structure of an IPv4 address?

An IPv4 address written in dotted decimal notation consists of four octets of eight bits (total length: 32 bit) expressed as decimal numbers in the range $[0, 255]$ which are separated by periods. Besides the periods, the total IPv4 address is split into two parts, the network identifier (ID) and the host ID. While the network ID starting from the left-most bit identifies the network where the host is located, the host ID identifies the host within the network.

Example: 11100011.01010010.10011101.1011001 $\widehat{=}$ 227.82.157.177

To express the network and host IDs in the dotted decimal notation as 32 bit addresses, they are padded with zeros.

Example: 277.0.0.0, 0.82.157.177

Note that the split between network and host ID happens on the binary level, i.e. it can also occur within an octet. When writing the network and host IDs as 32 bit addresses in the dotted decimal notation, octets can overlap.

Example: 11100011.01010010.10011101.1011001 $\widehat{=}$ 227.82.144.0, 0.0.13.177

4 4 What is an IPv4 network?

An IPv4 network is a group of computers which exchange messages using the internet protocol version 4.

4.5 What are network classes? Are they still relevant?

Network classes are a way to subdivide the available 2^{32} IP addresses such that they fit the requirements of large (10^2-10^6 hosts), medium (10^2-10^3 hosts) and small (≤ 250 hosts) organizations. There were five classes, however the most important ones are:

Class (fixed bits)	Fraction of IP address space	#Network ID bits	#Host ID bits	Usage
A (0)	1/2	8	24	Large organizations
B (10)	1/4	16	16	Medium organizations
C (110)	1/8	24	8	Small organizations

This decomposition was at the time simple (octet boundaries to distinguish the classes), flexible (three sizes of organizations with sufficient capacity), allowed for easy routing (check at most leading four bits) and reserved addresses for special purposes (classes D, E). However with the growth of the Internet, problems like a lack of internal address flexibility (large organizations get a large block of addresses without being able to match the structure of internal networks), inefficient use of address space (with only three block sizes IP address space is wasted) and the proliferation of router table entries (performance problems: Internet growth requires more entries for routers to handle routing) became apparent. Hence nowadays network classes are no longer used and classless addressing is the norm.

4.6 What is the purpose of a subnet mask?

While network classes split an IP address into two levels (network, host ID), a subnet mask adds a third level by grouping hosts into subnetworks by splitting the "classful" host ID into a subnet ID and a new host ID. This way organizations can group hosts, independent of and invisible to the public Internet, into subnets which reflect the internal structure of the organizations physical network. The subnets within the organizations network also do not contribute to an increasing number of entries in the routing tables of routers of the Internet, since they are perceived as belonging to a whole network. Also, no new IP addresses have to be requested as would be the case if multiple class C blocks were used. Given the subnetting, the mask allows routers to quickly determine the subnet address by means of masking. Using a mask instead of the prefix-length employed in the CIDR notation allows

TODO

4.7 How does subnetting work?

Explain it in the context of VLSM (variable length subnet masking) and CIDR (classless inter-domain routing)

TODO

Break a large network into smaller ones

Mention cheat table

Two devices in different networks separated by a router cannot communicate with each other Two devices in the same network separated by a router cannot communicate with each other

Supernetting: join smaller networks together

VLSM: create subnets of different sizes

Unicast traffic: Communication between two devices Multicast traffic: Way for devices to opt in to certain types of traffic Broadcast traffic: Communication between one and all other devices

Routers never forward broadcast messages

4.8 What is IPv6 and what is the approach of IPv6?

After IPv4, IPv6 is the next version of the internet protocol. It adresses consist of a 128 bit (IPv6) long binary number represented in the colon hexadecimal notation by eight groups of four hexadecimal digits separated

by colons. Leading zeros within the eight groups can be suppressed (zero suppression) and a single occurence of one or multiple groups filled by zeros can be replaced by two colons (zero compression). For example in fe80:4920::f529:c64:ab27/64 the third to fifth group are compressed, while in group seven one zero is suppressed. The most important changes from IPv4 to IPv6 are:

- Larger address space: $2^{32} \rightarrow 2^{128}$
- Hierarchical address space: More addresses allow more classes (groups of addresses) with the same size.
- Hierarchical assignment of unicast addresses: Global unicast address format which allows easy allocation.
- Better support for non-unicast addresses: Support for multicast is improved and anycast is added.
- Autoconfiguration and renumbering: Allow easier reconfiguration of hosts and renumbering of IP addresses in networks and subnetworks.
- New (IP) packet format: Header is streamlined and support for extending the header in case (IP) packets require more control information is provided
- Support for quality of servic (QoS): (IP) packets contain QoS features for a better support of multimedia and other applications.
- Security support: Authentication and encryption extension headers.
- Updated fragmentation and reassembly procedures: Fragmentation and reassembly of (IP) packets is changed to improve efficiency of routing.
- Modernized routing support: Support modern routing systems and allow expansion as the Internet grows.
- Transition capabilities: Support for transition from IPv4 to IPv6.

4.9 Problem 5: Subnetting

Decompose an arbitrary /24 network within the RFC1918 range into four parts by means of subnetting (VLSM/CIDR). The range of hosts refers to the number of IP addresses available for end devices.

Network: 10.157.80.0/24

- Subnet 1
  Network address: .0
  Subnet mask: /22
  Broadcast address: .63
  Range of hosts: [.1,.62]
- Subnet 2
  Network address: .64
  Subnet mask: /22
  Broadcast address: .127
  Range of hosts: [.65,.126]
- Subnet 3
  Network address: .128
  Subnet mask: /22
  Broadcast address: .191
  Range of hosts: [.129,.190]
- Subnet 4
  Network address: .192
  Subnet mask: /22
  Broadcast address: .255
  Range of hosts: [.193,.254]

5 Applications

5 1 Which application layer protocols are there?

Mention at least 15 and describe them shortly

Selection of application layer protocols:

- File transfer protocol (FTP): Transfer computer files from a server to a client on a computer network. Employs TCP on port 21.
- HyperText transfer protocol (HTTP(S)): Enables distributed, collaborative, hypermedia information systems by linking together resources which can be accessed by means of uniform resource locators (URLs). A secured variant is HTTPS. Both employ TCP and UDP. HTTP uses port 80 and HTTPS uses port 443.
- Teletype network (Telnet): Provides access to virtual terminals of remote systems on LANs or the Internet. It transmits all information including usernames and passwords in plaintext and is thus not suitable for security-sensitive applications. Employs TCP on port 23.
- Secure shell (SSH): Operate network services securely over an unsecured network. Mostly used for remote login and command-line execution. Employs TCP on port 22.
- Internet message access protocol (IMAP), Post office protocol v3 (POP3): Used by email clients to retrieve messages from a mail server over a TCP/IP connection. Both employ TCP. IMAP uses port 143 and POP3 uses port 110.
- Simple mail transfer protocol (SMTP): Used for electronic mail transmission, typically only for sending. Employs TCP on port 25.
- Transport layer security/Secure sockets layer (TLS/SSL): Provide communication security over a computer network. Since it is used by other protocols with their own ports, it cannot be associated with a specific port. One example is DNS+SSL/TLS employing TCP on port 853.
- Domain name system (DNS): Hierarchical and distributed naming system for devices to translate human-readable names into numerical IP addresses used by computers. Employs TCP and UDP on port 53.
- Internet relay chat (IRC): Text-based chat system for instant messaging which allows for group discussions in forums, one-on-one communication via private messages, data transfer and file sharing. Employs TCP and UDP on port 194.
- Dynamic host configuration protocol (DHCP): Automatically assigns IP addresses and other communication parameters to devices connected to the network using a client-server model. It employs UDP. The DHCP server uses port 67 and the DHCP client port 68.
- Routing information protocol (RIP): Interior gateway, distance-vector routing protocol employing the hop count as routing metric. Employs UDP on port 520.
- Border gateway protocol (BGP): Exterior gateway protocol to exchange routing and reachability information among autonomous systems (AS) on the Internet. It is a path-vector routing protocol which makes routing decisions based on paths, network policies or rule-sets configured by a network administrator. Employs TCP on port 179.
- Open shortest path first (OSPF): Interior gateway protocol operating within a single autonomous system (AS). It is a link-state routing protocol which makes routing decisions based on a topology map of the network constructed by gathering link-state information from available routers. It does not employ TCP, UDP and ports.
- Network time protocol (NTP): Performs clock synchronization between computer systems over packet-switched, variable-latency networks. Employs UDP port 123.
- Real-time transport protocol (RTP): Delivers audio and video over IP networks. Employs TCP and UDP on port 5004.
- Simple network management protocol (SNMP): Collect and organize information about managed devices on IP networks and modify that information to change device behavior. Employs only UDP on port 161 and TCP as well as UDP on port 162.

5 2 What is the identifier of an application?

The identifiers of applications are called sockets which are the combination of an IP address and a port number, separated by a colon as ⟨IP address⟩:⟨Port number⟩. Port numbers have a length of 16 bit. To send (application) messages, processes (instances of applications) specify the source and destination port to be used for communication. The ports are added to the TCP or UDP header, depending on which transport layer protocol is used. When the segments/datagrams are passed to the network layer, the protocol (TCP or UDP) used is specified in the protocol field of the (IP) packet header. Upon receiving (IP) packets, the network layer software inspects the protocol field of the (IP) packet and passes it on to TCP or UDP, if one of these protocols has been used. TCP/UDP receive the segments/datagrams and pass the (application) messages to the appropriate process based on the port number.

Multiplexing denotes the process in which segments/datagrams of different (application) messages labeled by the same IP address are perceived as a combination of indistinguishable (IP) packets that are send off separately. They appear to be indistinguishable on the network layer since the segments/datagrams port numbers are encapsulated and thus hidden. The reverse process of unpacking and identifying the process based on the port number is called demultiplexing.

There are three types of port number ranges:

- Well-known port numbers: Port numbers in the range $[0, 1\,023]$ are managed by the IANA and only used for the most universal TCP/IP applications which have been or likely will be standardized in the future. Sometimes called system port numbers.
- Registered port numbers: Port numbers in the range $[1\,024, 49\,151]$ can be requested from the IANA to be registered for a TCP/IP server application. This prevents applications from conflicting with each other.
- Private port numbers: Port numbers in the range $[49\,152, 65\,535]$ are neither reserved nor maintained by the IANA. Hence they can be used for any purpose without registration.

While servers must be approachable and thus use well-known or registered port numbers, clients use ephemeral port numbers which are only temporary. They are generated in a pseudo-random manner from a pool of reserved numbers in the range $[1\,024, 4\,999]$ while avoiding to use the same port number too quickly for another process.

On Windows, the command netstat -n can display the sockets of active connections.

5 3 What is the purpose of the DHCP and why is it needed?

The dynamic host configuration protocol (DHCP) uses a client-server model to automatically assign IP addresses and other communication parameters to devices connected to a network.

To add a host to a network, it has to be configured to communicate with other devices. For a large number of devices in a network, manual configuration is not feasible. Using the reverse address resolution protocol (RARP) on the data link layer is hardware-dependent and does not allow to transport configuration messages between networks. The bootstrap protocol (BOOTP) which operates on a higher layer overcomes RARPs shortcomings, but statically allocates IP addresses which is problematic for laptops moving between networks and restricted due to the exhaustion of the IP address space. DHCP solves these problems by automatically providing IP addresses upon connection to the network while limiting the time period for which the allocation is valid. Once this time period runs out, the old or a new IP address is provided by DHCP.

5 4 Describe in short how DHCP works

DHCP communicates connectionless via UDP using the port numbers 67 for the server and 68 for the client. Its operations fall in one of four phases:

1. Discover:
   - Client broadcasts DHCPDISCOVER message on the network subnet using 255.255.255.255 (limited broadcast) or the subnet broadcast address (directed broadcast)
   - Client may request a specific IP address in the DHCPDISCOVER message

2. Offer:
   - Several DHCP servers receive the DHCPDISCOVER message (IP address lease request)
   - DHCP servers create DHCPOFFER message (contain client MAC address, offered IP address, subnet mask, lease duration, IP address of DHCP server making the offer)
   - DHCP servers check if offered IP address is free with ICMP Echo message and reserve it for the client
   - DHCP servers send DHCPOFFER message to client as broadcast
3. Request:
   - Client receives DHCPOFFER messages from several DHCP servers and decides which to take (implementation-dependent, e.g. take offer from fastest responding server)
   - Client broadcasts DHCPREQUEST message to the servers requesting one offered address and rejecting the others
4. Acknowledge:
   - Rejected DHCP servers receive DHCPREQUEST message and wait a while before offering the rejected lease to another client
   - Accepted DHCP server checks if offered IP address is free
   - IP address taken: Accepted DHCP server sends DHCPNAK message (negative acknowledgement), client restarts with new DHCPDISCOVER message
   - IP address free: Accepted DHCP server sends DHCPACK message (acknowledgement) with IP address, lease length and other parameters as broadcast
   - Client checks if offered IP address free with ARP request
   - IP address taken: Client sends DHCPDECLINE message to accepted server, restarts with new DHCPDISCOVER message
   - IP address free: Host configured

5.5 Problem 6: Different servers in a local network

5.5.1 DHCP-server

1. The network components and end devices are configured as:

Computer	IP	MAC
PC0	DHCP	0060.707C.CCE6
PC1	192.168.12.1	00E0.8F0B.8D50
Laptop	DHCP	0060.7021.5BA8
Server0	192.168.12.250	0001.43BC.4A53
Server1	172.16.1.1	0060.5C0A.ADD5

2. To configure the DHCP server with an IP address in the range 192.168.12.50–192.168.12.100, one modifies under the tab Services>DHCP the Start IP address to 192.168.12.50, adapts the Subnet Mask to 255.255.255.0 and sets the Maximum Number of Users to 50. Note, that the DHCP server also has to be switched on under the tab.
3. If an IP address is provided by a DHCP server, the following commands show the respective lines
   - Windows: ipconfig /all shows the line “DHCP enabled...: Yes”
   - Linux: ip address shows the keyword “dynamic”

   Inside packet tracer, one can use ipconfig /release which will output the message “Port is not using DHCP.” if the IP address is not provided by a DHCP server. In contrast, if the IP address is provided by a DHCP server, the command returns the lines
   - IP Address...: 0.0.0.0
   - Subnet Mask...: 0.0.0.0
   - Default Gateway...: 0.0.0.0
   - DNS Server...: 0.0.0.0

This command also releases the IP address provided by the DHCP server. To request a new one, the command ipconfig /renew can be used.

4. To renew the dynamical IP addresses received by DHCP, one uses the following two commands:
   - ipconfig /release: Sends DHCPRELEASE message to DHCP server to release DHCP configuration and discard IP address configuration for all adapters. TCP/IP adapters configured to obtain IP addresses automatically are disabled, i.e. no new IP address are received from DHCP.
   - ipconfig /renew: Renew DHCP configuration for all adapters.

   In Linux, the respective commands are:
   - dhclient -r ⟨interface⟩: Release IP address. The term ⟨interface⟩ refers to one of the network interfaces shown by ip addr.
   - dhclient ⟨interface⟩: Request new IP address.

   By means of ipconfig /renew, one can also request an IP address from the DHCP server to replace a statically set IP address.
5. Adding another DHCP server with the IP address 172.16.1.1 providing IP addresses in the range 172.16.1.30–172.16.1.50 (Start IP address 172.16.1.30, Subnet Mask 255.255.0.0, Maximum Number of Users 20) has several effects:
   - Since the network ID of Server1s IP address is different from that of Server0 and PC1, they reside in different networks. Since the devices are only connected by a switch and not a router, they cannot communicate with each other.
   - Upon releasing and renewing the IP address, the DHCP clients PC0 and Laptop can switch networks, depending on which of the two DHCP servers offers are taken. As a result, PC0 may no longer be able to reach them by ping.
   - Strangely, sometimes the IP addresses received from Server1 arrive at the clients with the subnet mask 255.255.255.0. This can have real consequences, for example in the configuration

Computer	IP
PC0	172.16.1.31/255.255.0.0 (DHCP)
PC1	172.16.2.1/255.255.0.0
Laptop	172.16.1.32/255.255.255.0 (DHCP)

   PC0 can ping PC1, while the Laptop cannot (ARP request to infer MAC address uses wrong TGT IP address 0.0.0.0). Looking at the IP address of the Laptop, one would expect to be able to ping PC1.

5.5.2 Web-server

1. The network components and end devices are configured as:

Computer	IP	MAC
PC0	192.168.12.10	
www.netsec.at	192.168.12.251	
DNS	192.168.12.253	

   The DNS server is added later to reach the website by its hostname.
2. Using only the setup with PC0 and the web server without having specified a DNS server on PC0 immediately returns the message "Host Name Unresolved" in the browser when entering www.netsec.at.

   Specifying a DNS server on PCO and entering www.netsec.at in the browser triggers the creation of a DNS Query followed by an ARP request to find the server. PC0 attempts to find the DNS server five times until it displays the message "Host Name Unresolved".

   Connecting the DNS server to the switch without adding a pair of name and IP address, entering www.netsec.at in the browser triggers the creation of a DNS Query followed by an ARP request which finds the server. PC0 then sends off the DNS Query to the DNS server which returns without an IP address in the DNS Answer, since it has not been added yet.

To add an entry in a DNS server, one enters under the tab Services>DNS the name (www.netsec.at) and IP address (192.168.12.251) of the device and presses Add. Note, that the DNS server also has to be switched on under the tab.

Requesting www.netsec.at triggers the following process:

- Creation of a DNS Query followed by an ARP request to get the MAC address of the DNS server (192.168.12.253)
- DNS Query with field AME: www.netsec.at to DNS server which returns a DNS Answer with the fields NAME: www.netsec.at and IP: 192.168.12.251
- Creation of a TCP message followed by an ARP request to get the MAC address of the webserver (192.168.12.251)
- TCP three-way handshake:
  - TCP message from PC0 (port 1027) to www.netsec.at (port 80) with FLAGS: 0b00000010 where the SYN bit is set [Car21]
  - TCP message from www.netsec.at (port 80) to PC0 (port 1027) with FLAGS: 0b00010010 where the SYN and ACK bits are set [Car21]
  - TCP message from PC0 to www.netsec.at with FLAGS: 0b00010000 where the ACK bit is set [Car21]
- HTTP exchange:
  - HTTP REQUEST from PC0 (port 1027) to www.netsec.at (port 80) with TCP part containing the FLAGS: 0b00011000 where the ACK and PSH (push data out to sending TCP as soon as www.netsec.at can) bit are set [Car21; str11]
  - HTTP RESPONSE from PC0 (port 1027) to www.netsec.at (port 80) with TCP part containing the FLAGS: 0b00011000 where the ACK and PSH bit are set (no further data to send), ACKNOWLEDGEMENT NUMBER: 103 and an HTTP part requesting to close the connection
- TCP four-way handshake:
  - TCP message from PC0 (port 1027) to www.netsec.at (port 80) with FLAGS: 0b00010001 where the ACK and FIN (close connection) bits are set, SEQUENCE NUMBER: 103 and ACKNOWLEDGEMENT NUMBER: 472
  - TCP message from www.netsec.at (port 80) to PC0 (port 1027) with FLAGS: 0b00010001 where the ACK and FIN bits are set, SEQUENCE NUMBER: 472 and ACKNOWLEDGEMENT NUMBER: 104
  - TCP message from PC0 (port 1027) to www.netsec.at (port 80) with FLAGS: 0b00010000 where the ACK bit is set, SEQUENCE NUMBER: 104 and ACKNOWLEDGEMENT NUMBER: 472

When loading the image from the index.html, roughly the same procedure takes place, but without the ARP requests and with more HTML messages sent by www.netsec.at before requesting to close the connection. Every HTML message has the same ACKNOWLEDGEMENT NUMBER: 124 and an increasing SEQUENCE NUMBER. After every few HTML messages by www.netsec.at, PC0 sends a TCP message with the ACK bit set, the SEQUENCE NUMBER: 124 and the ACKNOWLEDGEMENT NUMBER set to the SEQUENCE NUMBER of the previously received HTML message.

5.6 What is the difference between HTTP and HTTPS?

The difference between the HyperText transfer protocol (HTTP) and HTTP secure (HTTPS) consists in the latter being encrypted using Transport layer security/Secure sockets layer (TLS/SSL).

5.7 What is the purpose of SSH and how is it different from Telnet?

The secure shell (SSH) serves to operate network services securely over an unsecure network. It is mostly used for remote login and command-line execution. While in Telnet all information including usernames and passwords is transmitted in plaintext, in SSH both ends of a communication channel use public-private key pairs to encrypt the network connection.

5.8 What is the purpose of FTP?

The file transfer protocol (FTP) serves to transfer computer files from a server to a client.

5.9 What is the purpose of encrypted protocols?

The purpose of encrypted protocols is to add a layer of security to a service. Examples for this are:

- HTTPS: HTTP+TLS/SSL on port 443
- SMTPS: SMTP+TLS/SSL on port 465
- DNS+TLS on port 853
- FTPS: FTP+TLS/SSL on ports 989, 990
- Telnet+TLS/SSL on port 992
- IMAPS: IMAP+TLS/SSL on port 993
- POP3S: POP3+TLS/SSL on port 995

6 Transport

6 1 What are ports?

Ports or port numbers are parts of the identifiers of processes discussed in section 5.2.

6 2 What is the difference between TCP, UDP and QUIC?

Characteristic	UDP	TCP
General description	Simple, high-speed, low-functionality "wrapper" that interfaces applications to the network layer and does little else.	Full-featured protocol that allows applications to send data reliably without worrying about network layer issues.
Protocol connection setup	Connectionless: Data is sent without setup	Connection-oriented: Connection must be established prior to transmission
Data interface to application	Message-based: Data is sent in discrete packages by the application	Stream-based: Data is sent by the application with no particular structure
Reliability and acknowledgements	Unreliable, best-effort deliver without acknowledgements	Reliable delivery of messages, all data is acknowledged
Retransmissions	Not performed. Application must detect lost data and retransmit if needed	Delivery of all data is managed and lost data is retransmitted automatically
Features provided to manage flow of data	None	Flow control using sliding windows*, window size adjustment heuristics, congestion avoidance algorithms
Overhead	Very low	Low, but higher than UDP
Transmission speed	Very high	High, but not as high as UDP
Data quantity suitability	Small to moderate amounts of data (up to a few hundered bytes)	Small to very large amounts of data (up to gigabytes)
Types of applications that use the protocol	Applications where delivery speed matters more than completeness, where small amounts of data are sent or where multicast/broadcast are used.	Most protocols and applications sending data that must be received reliably, including most file and message transfer protocols
Well-known applications and protocols	Multimedia applications, DNS, TFTP, RIP, BOOTP, DHCP, SNMP	DNS, FTP, BGP, Telnet, IRC, HTTP, POP, IMAP, SMTP

*The expression "sliding window" refers to the mechanism to control how much data the sender may transmit until the next acknowledgement by the receiver. Since the transport is stream-based, it does not make sense to acknowledge every single bit received. Instead, the segments get a sequence number attached to their last bit and the received bits with consecutive sequence numbers are counted. To not overwhelm the receiver, the

number of segments the sender is allowed to send is limited. The entire stream of segments can thus be split into four categories:

1. sent+acknowledged
2. sent+not acknowledged
3. not sent+allowed to be sent
4. not sent+not allowed to be sent

While the sender can move segments from category 3 to 2, the receiver can move them from category 2 to 1. After a given span of time, the receiver submits an acknowledgement to the sender containing the number of segments that have been acknowledged (moved from 2 to 1) since the last acknowledgement. Consequently, the sender adds the same number of segments from category 4 to category 3. Note that if a segment is dropped, a sequence number gets missing and category 3 is not extended. To handle this issue, a timer counts down since the last acknowledgement and if it runs out, the segments in category 2 are resend. The window in this description refers to an imaginary containement which slides over all segments and holds categories 2 and 3. To adjust the transmission speed, the window size can also be changed.

QUIC:

- Problems with TCP:
  - Head-of-line-blocking: Error on a connection is blocking operation →Further transfers are stopped, additional data sent until error is detected is blocked/flushed while error is corrected (affects HTTP/1, HTTP/2)
  - Multiple handshakes: TCP has little understanding of the requirements (e.g. encryption) data it transmits has →Protocols operating on top of TCP (e.g. TLS) perform handshakes as well
  - Decryption has to wait for partial packets (TLC operate on top of TCP)
- Goals of QUIC:
  - Replace TCP
  - Reduce connection and transport latency of connection-oriented web applications
  - Estimate bandwidth in each direction of a connection →avoid congestion with algorithms working in user space
  - Extend with forward error correction: Improve performance by fixing errors instead of retransmitting data
  - Avoid protocol ossification (loss of flexibility, extensibility, evolvability of network protocols)
- Improvements by QUIC:
  - Prevent head-of-line-blocking: Use UDP in multiple, separately flow controlled QUIC streams to retransmit lost data on the QUIC level →Transfers on other stream continue, additional data sent on other streams until error is detected are not blocked/flushed while erroneous stream is repaired
  - Combine handshakes during connection setup: Add parts from other handshake responses (e.g. TLS) into QUIC response →Rreduce overhead
  - Packets encrypted individually →Decryption does not have to wait for partial packets
  - Improve performance during network-switching events
  - QUIC can be implemented in application space
  - Concerns invalid: Middleboxes in network tuned to TCP and rate-limit/block UDP →Experiment: Only small number of connections blocked this way →Rapid fallback-to-TCP system

6 3 Why are there different transport protocols?

Since different applications have different priorities, different transport protocols with different benefits (reliability vs. transmission speed) are used. One example where both protocols are used is transport under DNS.

6.4 What is a three-way handshake?

Handshakes serve to establish rules for communication when a computer attempts to communicate with another device. Formally, a handshake is an automated process of negotiation between two participants that establishes the protocols of a communication link by exchanging information at the start of the communication, before full communication begins.

An example is the TCP three-way handshake used to establish a connection, which consists of three communication steps between the hosts H1 and H2:

1. H1 sends H2 a synchronize (SYN) message with sequence number x
2. H2 replies with a synchronize-acknowledgement (SYN-ACK) message, a sequence number y and an acknowledgement number $x + 1$
3. H1 replies with an acknowledgement (ACK) message and an acknowledgement number $y + 1$

SYN messages act as service requests, while ACK messages serve as confirmation that the message was received. The purpose of sequence and acknowledgement numbers is to prevent messages from earlier connections to interfere.

Other examples for handshakes are the TCP four-way handshake uses to terminate a connection or the lengthy TLS handshake to initiate a secured connection.

6.5 Which transport mechanism do all applications in section 5.1 use?

While some application protocols in section 5.1 use TCP, some UPD and some none of both, all of them use IP.

6.6 Problem 7: Networksniff with Wireshark & Linux command line

6.7 Wireshark on Linux

Since MakeMeAdmin did not grant me Administrator rights and Windows refused all combinations of usernames and passwords, I could not install Wireshark or the drivers to record in Wireshark portable on Windows. To go through the exercises nevertheless, I installed Wireshark on my own Linux machine and used it there.

1. Q: How does one filter the traffic of an IPv4 address?
   A: One can filter the traffic of an IPv4 address by using the protocol.element filter, e.g. ip.addr = 193.168.1.1..
2. Q: How does one filter for ICMP (pings)?
   A: One can filter for ICMP by using the protocol filter icmp.
3. While recording with Wireshark, open an HTTP website and filter for the IP address of the website and the protocol HTTP
   Following the response in [Dav17] website `http://lushbrightsilverjoke.neverssl.com/online/` is used as an http website.
   a. Q: What happens when following a TCP/HTTP stream?
      A: Since HTTP is not encrypted via TLS/SSL giving HTTPS, the streams show the transferred data in plaintext.
   b. Q: Do you notice anything about the HTTP packets?
      Due to the website examined, there is nothing special about the HTTP packets. However when examining a website that also provides an HTTPS version, ...
      TODO
4. Q: Filter for ARP and explain a few packets you see
   A: All ARP requests observed concern requests by the router or the laptop running Wireshark asking for the MAC address of each other. Within the screenshot, the following messages occur:
   - Router requests MAC address of laptop: 126, 874

- Laptop replies its MAC address to the router: 127, 875
- Laptop requests MAC address of router in a broadcast: 247
- Router replies its MAC address to the laptop: 248

Note that the broadcast by the laptop can be detected because Wireshark is running on the laptop initiating it.

5. Q: While recording with Wireshark, renew your dynamic IP address and filter for DHCP packets
   A: The DHCP packets encompass a DHCP Release message, as well as the cycle of messages exchanged when requesting and receiving a new IP address: DHCP Discover, DHCP Offer, DHCP Request and DHCP ACK.
6. What is the difference between Wireshark and tcpdump?
   A: There are many differences between Wireshark and tcpdump:
7. Use nslookup to resolve several websites (e.g. bearingpoint.com) with the google DNS server 8.8.8.8
   a. Q: What do you notice about the DNS packets?
      A: There are a few noteworthy points about the DNS packets:
      - The DNS packet lists as source the laptop and as destination the google DNS server.
      - There are two replies by the DNS server for every website, one with the IPv4 address (record type A) and one with the IPv6 address (record type AAAA).
      - The DNS requests address the port 53 and are thus not encrypted.
   b. Q: Can one also follow a stream here?
      A: Yes, one can follow a UDP stream.

6.8 tcpdump on Linux

1. Q: How does one filter the traffic for a specific interface? Does one always have to specify an interface?
   A: To filter the traffic for a specific interface, one uses the -i option, e.g. sudo tcpdump -i wlp0s20f3. If no interface is specified, tcpdump
2. Q: How does one filter the traffic for an IPv4 address?
   A: To filter the traffic for an IPv4 address, one uses the host option, e.g. sudo tcpdump host 192.168.1.1
3. Q: How does one filter the traffic for two IPv4 addresses?
   A: To filter the traffic for two IPv4 addresses, one can use the or option to combine multiple filters, e.g. sudo tcpdump host 192.168.1.1 or host 8.8.8.8. Note that there also other logical operators like and and not. One can also filter separately for source and destination IP addresses using the src and dst options, e.g. sudo tcpdump src 192.168.1.4 or dst 8.8.8.8.
4. Q: How does one filter for ICMP (pings)?
   A: To filter for ICMP, one uses the protocol name as filter, e.g. sudo tcpdump icmp. Besides filtering for a protocol, one can use the port option to filter for a port, e.g. sudo tcpdump port 80.
5. Q: What do the parameters -v, -vv, -vvv accomplish?
   A: These parameters produce from -v to -vv to -vvv increasingly more verbose output.
6. Q: How does one save the tcpdump as a file?
   A: To save a tcpdump into a file, one uses the write option -w, e.g. sudo tcpdump -i wlp0s20f3 -w recording.pcap. To read such a file, one uses the read option -r, e.g. tcpdump -r recording.pcap. Filtering while reading is also possible.
7. Save a tcpdump with network traffic (should contain SSH, HTTP/HTTPS, ICMP →SSH to another device, open websites, ping another device) in a file and open it with Wireshark
   Connecting via SSH with my old institute in Stuttgart gives the SSH traffic, looking up some tools from Kali Linux on `https://www.kali.org/tools/` produces only HTTPS traffic and pinging my home-router contributes the ICMP traffic.

6.9 Networking tools on Linux

- netstat/ss
- tcpdump
- ping
- traceroute/tracepath
- tracepath

- nmap
- dig/nslookup
- host
- hostname
- arp
- ifconfig
- route
- iwconfig
- ip
- mtr
- curl
- wget
- whois
- ssh
- scp, rsync
- nc/netcat/ncat
- lwconfig
- finger
- ethtool
- tshark
- tcpflow
- rarp
- nameif
- plipconfig
- slattach
- mii-tool
- iptunnel
- ipmaddr
- ettercap
- dnsenum
-
-
-

6 10 Multitasking?

How can a computer access three websites simultaneously and distinguish which traffic (bits, bytes) belongs to which website? Watch the video `https://www.youtube.com/watch?v=RDotMcs0Erg`.

The access to three different websites can be distinguished by different source port numbers while all use the same destination port number 80 for HTTP or 443 for HTTPS.

6 11 What is a PDU?

How does the PDU on different layers look like?

Protocol data units (PDUs) are messages used by protocols to communicate between processes on different devices on the same layer of the OSI model. Since all layers but the first are logical, communication runs through all layers below the sending layer. While a layer above the current one passes data to it, the layer below the current one provides services.

Starting at a layer N, a PDU is the message which implements the protocol at that layer. When this N PDU is passed down to the next layer, it becomes the data that the $N-1$ protocol services and is thus called $N-1$ service data unit (SDU). To fulfill its task of transporting the $N-1$ SDU, layer $N-1$ performs data encapsulation by putting the $N-1$ SDU as data in its own $N-1$ PDU adding headers and trailers before and after the SDU. The $N-1$ PDU is then passed to layer $N-2$, where it is encapsulated and passed on until the physical layer is reached. At the receiver, the PDUs are de-encapsulated and the SDUs passed to the layer

above. A PDU is thus the message of the current layer, possibly encapsulating the PDU of the previous layer with headers and trailers.

Specifically in the OSI model, the PDUs are named:

Layer	PDU	SDU
Application (7)	(Application) message	–
Transport (4)	(TCP) segment/(UDP) datagram	(Application) message
Network (3)	(IP) packet	(TCP) segment/(UDP) datagram
Data link (2)	(Ethernet) frame	(IP) packet
Physical (1)	Bits/bytes	(Ethernet) frame

6 12 Problem 8: Large local network

6 12 1 Multiple webserver, FTP server

The network components and end devices are configured as follows:

Computer	IP	MAC
PC0	DHCP	0001.432D.C917
PC1	192.168.12.1	0002.4A45.5C47
PC2	192.168.12.10	0030.F25C.7B91
PC3	192.168.12.20	0009.7C64.4196
PC4	DHCP	00E0.8F84.B28E
Laptop	DHCP	0060.47CA.8719
DHCP	192.168.12.250	0007.EC5D.E747
DNS1	192.168.12.253	0004.9AB1.24B2
www.bearingpoint.com	192.168.12.120	000C.CF63.97C1
www.netsec.at	192.168.12.251	00D0.BA83.E44A
ftp.filesharing.org	192.168.12.130	000C.CF58.D51B

To check that every client can reach both webservers, both domain names www.bearingpoint.com and www.netsec.at are entered in the webbrowser of each client and display both websites. To verify that the FTP server works, on each client a paragraph of a poem is created and saved (Ctrl+s) with the text editor (three paragraphs from “In Flanders fields”, one paragraph from “Der Erlkönig” and one paragraph from “Der Zauberlehrling”). After accessing the FTP server with ftp ftp.filesharing.org and entering a pair of username and password, the files are uploaded using put ⟨filename⟩ and downloaded using get ⟨filename⟩. To check if the download worked, the received files are opened (Ctrl+o) in the text editor.

Since opening a website encompasses a DNS request, a three-way handshake and a HTTP request, all three requests will be described as parts of the complete process. To include ARP requests, the ARP table of PC0 is cleared with arp -d before requesting a website. As an example, PC0 enters the URL www.netsec.at in its browser.

- ARP request: PC0 sends off an ARP request (DST MAC: FFFF.FFFF.FFFF, TGT IP: IP_{DNS1}) as broadcast to infer the MAC address of DNS1 (MAC_{DNS1})
- DNS request: PC0 requests the IP address of the webserver www.netsec.at
  1. PC0 sends off a DNS Query (DST MAC: MAC_{DNS1}, DST IP: IP_{DNS1}, UDP, DST port: 53, $Name_Q$: www.netsec.at)
  2. Path: PC0, switch0, switch1, DNS1
  3. DNS1 replies with DNS Answer (DST MAC: MAC_{PC0}, DST IP: IP_{PC0}, UDP, SRC port: 53, $Name_A$: www.netsec.at, IP_A: IP_n)
  4. PC0 receives IP_n
- ARP request: PC0 sends off an ARP request (DST MAC: FFFF.FFFF.FFFF, TGT IP: IP_n) as broadcast to infer the MAC address of www.netsec.at (MAC_n)

- Three-way handshake:
  1. SYN: PC0 sends off a TCP message (DST MAC: MAC_n, DST IP: IP_n, DST Port: 80, TCP FLAGS: 0b00000010) to www.netsec.at with the SYN bit set
  2. Path: PC0, switch0, switch1, www.netsec.at
  3. SYN-ACK: www.netsec.at sends off a TCP message (DST MAC: MAC_{PC0}, DST IP: IP_{PC0}, SRC Port: 80, TCP FLAGS: 0b00010010) to www.netsec.at with the SYN and ACK bit set
  4. Path: www.netsec.at, switch0, switch1, PC0
  5. ACK: PC0 sends off a TCP message (DST MAC: MAC_n, DST IP: IP_n, DST Port: 80, TCP FLAGS: 0b00010000) to www.netsec.at with the ACK bit set

  Right after the third step of the TCP three-way handshake is complete, PC0 sends off a HTTP request.
- HTTP request:
  1. PC0 sends a HTTP request (DST MAC: MAC_n, DST IP: IP_n, DST port: 80, TCP FLAGS: 0b00011000) to www.netsec.at with the ACK and PSH (push data out to sending TCP as soon as www.netsec.at can) bit set
  2. www.netsec.at sends two Variable Sizes PDUs (DST MAC: MAC_{PC0}, DST IP: IP_{PC0}, SRC port: 80, TCP FLAGS: 0b00010000) to PC0 with the same ACKNOWLEDGEMENT NUMBER (103) and increasing SEQUENCE NUMBER (1, 557) and the ACK bit set
  3. www.netsec.at sends a HTTP response (DST MAC: MAC_{PC0}, DST IP: IP_{PC0}, SRC port: 80, TCP FLAGS: 0b00011000) to PC0 with ACK and PSH bit set (no further data to send), the ACKNOWLEDGEMENT NUMBER of the previous Variable Sizes PDUs (103) and increased SEQUENCE NUMBER (1113)

  With this HTTP response, PC0 has received the website. However the process is not yet over since TCP still has to close the connection.
- Four-way handshake:
  1. ACK-FIN: PC0 sends a TCP message (DST MAC: MAC_n, DST IP: IP_n, DST port: 80, TCP FLAGS: 0b00010001, SEQUENCE NUMBER: 103, ACKNOWLEDGEMENT NUMBER: 1395) to www.netsec.at with ACK and FIN (close connection) bit set
  2. ACK, FIN: www.netsec.at sends a TCP message (DST MAC: MAC_{PC0}, DST IP: IP_{PC0}, SRC port: 80, TCP FLAGS: 0b00010001, SEQUENCE NUMBER: 1395, ACKNOWLEDGEMENT NUMBER: 104) to PC0 with the ACK and FIN bit set
  3. ACK: PC0 sends TCP message (DST MAC: MAC_n, DST IP: IP_n, DST port: 80, TCP FLAGS: 0b00010000, SEQUENCE NUMBER: 104, ACKNOWLEDGEMENT NUMBER: 1395) to www.netsec.at with ACK bit set

- Q: Which transport protocol do DHCP, DNS, HTTP and HTTPS use?
  A: DHCP and DNS use UDP, while HTTP and HTTPS use TCP.
- Q: Prove with PDUs which ports are used by DNS, HTTP, HTTPS and DHCP
  A: By requesting a website with http:// and https:// and looking at the DST Ports in the UDP and TCP headers, one can infer the port 53 for DNS, port 80 for HTTP and port 443 for HTTPS. For the DHCP port one can execute ipconfig /release and ipconfig /renew on a client and observe the ports in the UDP header as 68 for the client and 67 for the server.
- Q: Are default gateways required?
  A: In the current setup default gateways are not required, since all devices reside in the same network.

6.12.2 DNS forwarding

The two additional DNS servers are configured as follows:

Computer	IP	MAC
org-dns	192.168.12.150	000A.4177.958B
com-dns	192.168.12.151	00E0.F79B.8B51

To accomplish DNS forwarding from DNS1, the identifier of the top-level domain (e.g. com, org) has to be assigned by name to an authoritative name server. For this purpose, the following resource records are added to DNS1:

name	type	detail
com	NS	com-dns
com-dns	A Record	192.168.12.151
org	NS	ord-dns
org-dns	A Record	192.168.12.150

This forwards all domain names ending in com and org from DNS1 to the right DNS server specified by its IP address. Since www.netsec.at does not belong to any of the two domains, it remains at DNS1. The domain name www.bearingpoint.com is added to com-dns, while ftp.filesharing.org is added to org-dns. Each of the additional DNS servers also receive their respective NS and A record.

Note that to assign authoritative name servers for second level or further subdomains, the second+top-level domain (e.g. example.com) or some subdomain (e.g. blog.example.com) have to be used in the name field.

When testing DNS forwarding, it is important to clear the DNS cache under the tab Services>DNS by clicking on the DNS Cache button and choosing Clear Cache. Otherwise DNS1 may already know the IP address for the requested domain name and directly answer the DNS Query without passing the DNS Query on to the authoritative DNS server.

- Q: Which DNS record is used to forward DNS Queries to other DNS servers?
  A: To forward DNS Queries, the DNS record of type NS is used, which delegates a DNS zone to use the given authoritative name servers.
- DNS record types:
  - Major record types:
    - Address (A) record: Provides the IPv4 address for a given hostname. It is used directly during name resolution.
    - AAAA record: Provides the IPv6 address for a given hostname. It is used directly during name resolution.
    - Canonical name (CNAME) record: Used to point a domain name (alias) to another domain (canonical name). It is used to run multiple subdomains for different purposes on the same server (e.g. use CNAME to point ftp.example.com and www.example.com to the same subdomain example.com which points to hte IP address using an A record).
    - Name server (NS) record: Specifies the authoritative DNS server for a domain. If the direct DNS server has no record on a requested domain name, it forwards the request to the corresponding authoritative DNS server for the given domain which either answers it or points to an authoritative DNS server for the subdomain. It serves as a branch of the DNS tree while A records can be seen as leaves. Usually, multiple name servers are specified for a domain.
    - Mail exchange (MX) record: Points to a mail server where emails for a given domain should be routed to.
  - Other record types:
    - Start of authority (SOA) record: Stores admin information (e.g. email address of admin, last time the domain was updated) about a domain.
    - Text (TXT) record: Enables owner of domain to store text values in the DNS record. Used to verify ownership of a domain.
    - Pointer (PTR) record: Provides domain name for reverse lookup, i.e. it is a reverse A record providing a domain name for a given IP address.
    - SRV record: Store IP address and port for specific services.
    - CERT record: Stores public key certificates.
    - DCHID record: Stores information related to DHCP.
    - Delegation name (DNAME) record: Points all subdomains for the alias also to the canonical domain name, i.e. it makes the alias transitive (e.g. secondsite.com →example.com implies staff.secondsite.com →staff.example.com).

6 12 3 Mail server

The two additional email servers are configured as follows:

Computer	IP	MAC
bearingpoint.com	192.168.12.240	0030.F207.54CB
mail.netsec.at	192.168.12.220	0007.EC21.6576

Server configuration:

Entry	Bearingpoint	Netsec
Domain name	bearingpoint.com	mail.netsec.at
Username	jan.lotze	tony.stark
Password	test1234	jarvis

Client configuration:

Entry	Bearingpoint	Netsec
Your name	Jan Lotze	Tony Stark
Email address	jan.lotze@bearingpoint.com	tony.stark@mail.netsec.at
Incoming mail server	bearingpoint.com	mail.netsec.at
Outgoing mail server	bearingpoint.com	mail.netsec.at
username	jan.lotze	tony.stark
password	test1234	jarvis

Beside the clients and servers, also the DNS servers get additional records such that server and client can be configured without using IP addresses:

Location	DNS server	Name	Type	Detail
Bearingpoint	com-dns	bearingpoint.com	A Record	192.168.12.240
Netsec	DNS1	mail.netsec.at	A Record	192.168.12.220

Since the email transfer from PC1 (Jan) to the email server bearingpoint.com is more involved than the one from PC4 (Tony), it is described here:

- ARP request by PC1: Get the MAC address of DNS1 since it is the primary DNS server
- DNS Query by PC1 from DNS1: Get the IP address for the domain name bearingpoint.com
  - ARP request by DNS1: Get the MAC address of com-dns since it is the authoritative name server for the com domain
  - DNS Query by DNS1 from com-dns: Get the IP address of the domain name bearingpoint.com
  - ↩ DNS answer by com-dns for DNS1: Provide the IP address for the domain name bearingpoint.com
- ↩ DNS answer by DNS1 for PC1: Provide the IP address for the domain name bearingpoint.com
- ARP request by PC1: Get the MAC address of bearingpoint.com from its IP address
- TCP three-way handshake between PC1 and bearingpoint.com: Establish connection
- SMTP messages between PC1 and bearingpoint.com: Send data
- TCP four-way handshake between PC1 and bearingpoint.com: Close connection

For PC4 (Tony), the process is similar, except that DNS1 already has the IP address of the email server mail.netsec.at and thus no additional DNS Query is required.

After the email has been transferred from PC1 (Jan) to bearingpoint.com, the process of receiving the email is shorter due to ARP and DNS caching:

- DNS Query by PC1 from DNS1: Get the IP address for the domain name bearingpoint.com
- ↩ DNS answer by DNS1 for PC1: Provide the IP address for the domain name bearingpoint.com
- TCP three-way handshake between PC1 and bearingpoint.com: Establish connection
- SMTP messages between PC1 and bearingpoint.com: Send data
- TCP four-way handshake between PC1 and bearingpoint.com: Close connection

This procedure is the same for PC4 (Tony) with the email server mail.netsec.at, since DNS caching does not shorten the procedure.

- Q: Do the IP addresses change? Which protocols are used? Which ports are used? What is standing in the email client if an email was correctly sent off and what if sending failed?
  A: Considering the different number of DNS Queries required to send an email to the respective email servers, one can argue that for PC1 (Jan) the IP addresses involved change during all DNS Queries, while they stay the same for PC4 (Tony). First, PC1 (Jan) requests the IP address of bearingpoint.com from DNS1 with $IP_{SRC} = IP_{PC1}$ and $IP_{DST} = IP_{DNS1}$. DNS1 in turn requests the IP address of bearingpoint.com from com-dns with $IP_{SRC} = IP_{DNS1}$ and $IP_{DST} = IP_{com}$. When com-dns returns the IP address of bearingpoint.com, $IP_{SRC} = IP_{com}$ and $IP_{DST} = IP_{DNS1}$ and lastly when DNS1 answers PC1, $IP_{SRC} = IP_{DNS1}$ and $IP_{DST} = IP_{PC1}$.
  The Protocols and ports used while sending and receiving an email are:

Protocol	Ports	Purpose
ARP	–	get MAC address from IP address
Ethernetll	–	contain MAC address
DNS/UDP	53	get IP address from hostname
IP	–	contain IP address
SMTP/TCP	25	send email
POP3/TCP	110	receive email

- The process of sending a mail from PC1 (Jan) to PC4 (Tony):
  - PC1 sends email to bearingpoint.com:
  - ARP request by PC1: Get the MAC address of DNS1 since it is the primary DNS server
  - DNS Query by PC1 from DNS1: Get the IP address for the domain name bearingpoint.com
    - ARP request by DNS1: Get the MAC address of com-dns since it is the authoritative name server for the com domain
    - DNS Query by DNS1 from com-dns: Get the IP address of the domain name bearingpoint.com
    - ↩ DNS answer by com-dns for DNS1: Provide the IP address for the domain name bearingpoint.com
  - ↩ DNS answer by DNS1 for PC1: Provide the IP address for the domain name bearingpoint.com
  - ARP request by PC1: Get the MAC address of bearingpoint.com from its IP address
  - TCP three-way handshake between PC1 and bearingpoint.com: Establish connection
  - SMTP messages between PC1 and bearingpoint.com: Send data
  - TCP four-way handshake between PC1 and bearingpoint.com: Close connection

  - bearingpoint.com sends email to mail.netsec.at:
  - ARP request by bearingpoint.com: Get the MAC address of DNS1 since it is the primary DNS server
  - DNS Query by bearingpoint.com from DNS1: Get the IP address for the domain name mail.netsec.at
  - ↩ DNS answer by DNS1 for bearingpoint.com: Provide the IP address for the domain name mail.netsec.at
  - ARP request by bearingpoint.com: Get the MAC address of mail.netsec.at from its IP address
  - TCP three-way handshake between bearingpoint.com and mail.netsec.at: Establish connection
  - SMTP messages between bearingpoint.com and mail.netsec.at: Send data
  - TCP four-way handshake between bearingpoint.com and mail.netsec.at: Close connection

  - PC4 requests email from mail.netsec.at:
  - ARP request by PC4: Get the MAC address of DNS1 since it is the primary DNS server
  - DNS Query by PC4 from DNS1: Get the IP address for the domain name mail.netsec.at
  - ↩ DNS answer by DNS1 for PC4: Provide the IP address for the domain name mail.netsec.at
  - ARP request by PC4: Get the MAC address of mail.netsec.at from its IP address
  - TCP three-way handshake between PC4 and mail.netsec.at: Establish connection
  - SMTP messages between PC4 and mail.netsec.at: Send data
  - TCP four-way handshake between PC4 and mail.netsec.at: Close connection

7 Network address translation (NAT)

Watch the videos in
https://www.youtube.com/playlist?list=PLCb8EhYsrW_s-gIU867OrVcDoAFEDjhDl

7 1 Why is NAT needed/what is the basic idea behind it?

Network address translation (NAT) is needed to translate private IP addresses into public IP addresses. It allows to delay running out of IPv4 addresses by having large numbers of hosts with private IP addresses in the same network share a small number of public IP addresses. Furthermore, different networks can use the same private IP addresses for their hosts and thus preserve IP addresses. NAT also allows the distinction between private and public addresses which adds a layer of protection against cyber threats since hosts with private IPs cannot be reached from outside the network.

A basic NAT implementation starts with the network of an organization set up with IP addresses in one of the private addressing ranges. One or more public IP addresses are also assigned to the organization and NAT-capable routers connect the local network with the public Internet. Comparing this setup to a telephone system, the public IP addresses are "outside lines", private IP addresses are "internal extensions " and the NAT routers act as telephone system computer or receptionist. The NAT routers map private to public IP addresses and guide traffic by IP routing and modifying IP packets, thus translating private into public IP addresses and vice versa.

Advantages of NAT are:

- Public IP address sharing: A large number of hosts can share a small number of IP addresses which saves money and conserves IP addresses.
- Easier expansion: Since devices in a local network are privately addresses and not each needs a public IP address, it is easy to add new clients to the network.
- Greater local control: Administrators get the benefits of a private network, but still can connect to the Internet.
- Greater flexibility in ISP service: Changing the organizations ISP is easier, since only public IP addresses change while private ones stay the same.
- Increased security: NAT provides indirection, i.e. clients cannot be accessed from outside the network since they do not have a public IP address.
- (Mostly) transparent: NAT implementations are transparent since changes occur only at a few routers.

Disadvantages of NAT are:

- Complexity: NAT adds one more complexity to setting up and managing a network. It also makes troubleshooting more confusing due to address substitutions.
- Problems due to lack of public addresses: Certain functions will not work properly due to lack of public IP addresses on client hosts.
- Compatibility problems with certain applications: Since NAT modifies IP header fields but not application data, tools like FTP which pass IP addresses and port numbers in commands have to be handled specially and some applications may not work.
- Problems with security protocols: Protocols like IPSec are designed to detect modifications to headers and reacts to changes NAT makes since it cannot distinguish them from malicious hacking. Combining NAT with IPSec is possible but complicated.
- Poor support for client access: Legitimate access to clients in a local network from outside is made difficul by NAT, "peer-to-peer" applications are harder to set up with NAT and something to be accessed from the public Internet (e.g. organization website) is set up without NAT.
- Performance reduction: For each (IP) packet passing between private network and Internet an address translation and other work (e.g. recalculation of the header checksums) has to be performed. While each translation takes little effort, some overall performance is lost.

In order to describe what NAT and its different variants do, it is useful to introduce two types of classification to capture the type of IP address discussed. One classification is the distinction between the locations of the devices owning the addresses:

- Inside address: IP address of a device inside the local network
- Outside address: IP address of a device outside the local network, i.e. in the Internet

The other classification concerns the perspective from which the devices owning these addresses is seen:

- Local address: IP address of a device viewed from inside the local network
- Global address: IP address of a device viewed from outside the local network

All four combinations of the two different classifications lead to the following types of IP addresses:

- Inside local IP addresses: IP addresses of a device inside the local network seen from inside the local network. Outgoing traffic starts from such an IP address before NAT and incoming traffic ends at such an IP address after NAT.
- Inside global IP addresses: IP addresses of a device inside the local network seen from outside the local network. Outgoing traffic comes from such an IP address after NAT and incoming traffic is sent off to such an IP address before NAT.
- Outside local IP addresses: IP addresses of a device outside the local network seen from inside the local network. Outgoing traffic is sent off to such an IP address before NAT and incoming traffic comes from such an IP address after NAT.
- Outside global IP addresses: IP addresses of a device outside the local network seen from outside the local network. Outgoing traffic is sent off to such an IP address after NAT and incoming traffic comes from such an IP address before NAT.

In practice, NAT uses a translation table which maps inside local IP addresses (regular IP address inside local network) to inside global IP addresses (public IP addresses used for external communication). It may also contain mapping between outside global IP addresse and outside local IP addresses, if necessary. Mappings between inside local and inside global IP addresses are called source NAT, while mappings between outside local and outside global IP addresses are called destination NAT. The translation table of a NAT router countains these terms inside local, inside global, outside local and outside global.

7 2 What is static NAT and what is it used for?

Give a short explanation and provide a usecase.

In static NAT, mappings in the translation table are permanent, fixed relationships between local and global IP addresses for inside or outside devices. Example: If the inside local IP address 10.0.0.207 is statically mapped to the inside global IP address 194.54.21.10, every IP packet from the inside device with 10.0.0.207 outside the network will show the IP address 194.54.21.10. Static NAT is used for devices that need to be represented with the same public IP address outside of the network (e.g. servers in local networks provided by companies to external clients). It is also used to allow inbound traffic to a particular device. In any case, static NAT requires manual setup and maintenance, and inside global IP addresses used in static NAT are not available for IP sharing on the internal network.

Inbound NAT or destination NAT, where an outside device request communication with an inside device, is straightforward via the corresponding inside global IP address.

7 3 What is dynamic NAT and what is it used for?

Give a short explanation and provide a usecase.

In dynamic NAT, mappings in the translation table are generated automatically by the NAT router as needed and discarded afterwards. Commonly a pool of inside global IP addresses are shared between a large number of inside devices. If necessary, a pool of outside local IP addresses is reserved for outside devices to use. Example: Assume a pool of inside global IP addresses in the range 194.54.21.1 to 194.54.21.20. When the inside device with 10.0.0.207 sends a request out of the network, it is mapped to one of the 20 IP addresses in the pool. Once the session is over, the assigned inside global IP address is discarded and returned to the

pool. The next request of the inside device with 10.0.0.207 may provide another inside global IP address. Dynamic NAT is used for regular clients to facilitate sharing of inside global (public) IP addresses. While it is more complicated than static NAT, it is automatic.

Note that static and dynamic NAT can be mixed. One just has to ensure that static mappings do not overlap with the pool used for dynamic assignment.

Inbound NAT or destination NAT is more involved with dynamic NAT than it is with static NAT. The problem is that the inside local IP address is hidden and addressing an inside global IP address which is not statically connected to the intended inside local IP address does not lead anywhere. A solution to this problem is to combine name resolution via DNS with dynamic NAT which proceeds as follows:

- The outside device sends a DNS request using the name of an inside device (e.g. www.ilikenat.com) it wishes to reach.
- The DNS server for the internal network resolves the name (e.g. www.ilikenat.com) into an inside local IP address for the device corresponding to this DNS entry.
- The inside local IP address is passed to NAT, to create a dynamic mapping between the inside local IP address of the device requested from outside and an inside global IP address. This mapping is added to the NAT routers translation table.
- The DNS server returns the inside global (public) IP address to the outside device.

7.4 What is PAT and what is it used for?

Give a short explanation and provide a usecase.

One issue with NAT is that the inside global (public) IP addresses available can run out and remaining hosts requesting one cannot access the Internet. Port address translation (PAT) alleviates this situation by translating in addition to the IP address also the port numbers used by TCP and UDP. This way, multiple hosts can share the same IP address. It requires a router capable of mapping IP address and port numbers. Disadvantages are greater complexity and potential compatibility issues with applications which contain the IP address or port numbers also on higher layers (e.g. FTP). Note that PAT also assumes traffic using TCP and UDP as it requires port numbers. If neither protocol is used, there is no port number and PAT does not work. Example: In case of unidirectional NAT, for outgoing traffic the source port is translated, while for incoming traffic the destination port is translated. PAT is used by home routers.

Inbound NAT or destination NAT in the context of PAT is called port-forwarding, where traffic from outside the local network addressed to an inside global IP address and a specific port is forwarded to a designated device. In other words, PAT maps a specific outside port to an inside local IP address and the given port.

8 Router

8.1 What is the purpose of a router?

While switches connect devices within a network and allow direct delivery, routers connect different networks and enable indirect delivery, which is also called routing.

Starting from the sender, the IP address of the destination is known, however the way there or the actual location of the destination are unknown. After handing off the IP packet to the next router, it is passed from router to router in a process called next-hop routing, until it reaches its destination network, where it is directly delivered.

Decisions where to send the IP packet are made locally, i.e. first the sender uses its information to pass the IP packet to the correct router they are attached to. Then, the router determines based on the IP address of the destination where the IP packet should hop to. This process repeats at each router along the IP packets path until it arrives in the correct network. Each router only need to know the next step, rather than the exact route.

A hop consists of the router examining the destination IP address in the header of the IP packet on the network layer, deciding which device to send the IP packet next and sending it off via the next network to the next router or destination host. For this purpose the router either has a record of the MAC addresses of the routers it is connected to, or it uses an ARP request to determine them.

To decide where to send an IP packet next, a router uses an internal routing table which provides a mapping between different network IDs and the routers it is connected to. Each routing entry contains the information that if the destination of the IP packet is in the following network, the next hop should take it to the following device. Hence the time to decide where to send the IP packet is shorter, the less entries a routing table contains. With the hierarchical structure of an IP address in classless addressing, routers aggregate routes into supernets to reduce the router table size. Addressing a network is the first step to subsequently address any subnet and addressing one of these allows to subsequently address any subsubnet. Since a network and its first subnet share the same network ID, the routers first check the subnet since it covers less hosts. Note that a routing table also contains information it has learned about more distant networks.

The purpose of routing protocols is to exchange information about routes and networks between routers to determine the best routes.

8.2 What is a default/standard gateway?

The term gateway used to refer to the device now called router. Nowadays gateways are different devices working on all seven layers. A default/standard gateway is the router which provides default routing functions for a particular device. When one device wants to send an IP packet to another device it cannot see on its local network, it sends it to the default gateway which takes care of the routing.

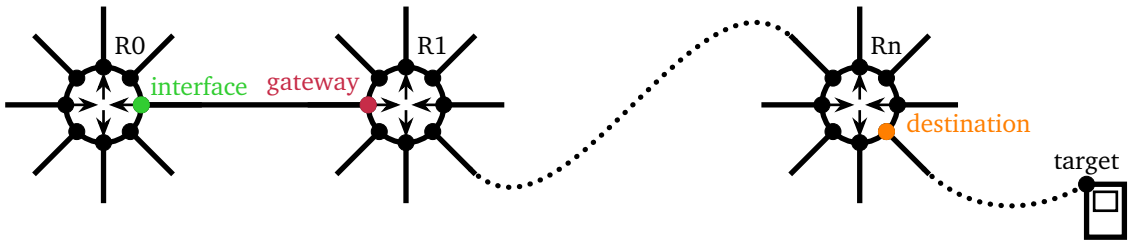
8.3 When does a computer send a packet directly to a router?

A computer sends an (IP) packet directly to a router if the receiving router is the default/standard gateway or it provides a service such as dynamic IP addresses by means of DHCP.

8.4 How are static routes built?

The static route configured in a given router (R0) consist of a destination (network address+subnet mask), a gateway (on-link or IP address of the next router interface (IR1) which can route the traffic further) and an

interface (IR0: IP address of R0s interface connecting to IR1). Below is an illustration of these three parts of a static route:



8 5 How can one connect from a computer to a router?

ssh/telnet/console

Ways to connect with a router:

- Web based interface (may allow to enable ssh and telnet access)
- Telnet: telnet ⟨IP address⟩ (unsafe since plaintext)
- SSH: ssh ⟨user⟩@⟨IP address⟩ (safe since encrypted)

Setup a router in packet tracer to allow remote access:

-
-
-

Setup a switch in packet tracer to allow remote access:

-
-
-

8 6 Problem 9: Introduction to routing

Configuration of a router [IPC18]:

- Set up the ⟨IP address⟩ of an ⟨Interface⟩ of a router:
  - enable>configure terminal
  - interface ⟨Interface⟩, e.g. interface GigabitEthernet 0/0 (use tab for autocompletion, be aware of whitespace)
  - ip address ⟨IP address⟩ ⟨Subnetmask⟩, e.g. ip address 192.168.0.10 255.255.255.0
- Activate the ⟨Interface⟩ of a router:
  - enable>configure terminal
  - interface ⟨Interface⟩
  - no shutdown
- Set up up a DHCP server at a router:
  - enable>configure terminal
  - ip dhcp pool ⟨name⟩
  - network ⟨network IP address⟩ ⟨Subnetmask⟩
  - default-route ⟨router IP address⟩
  - dns-server ⟨DNS server IP address⟩, exit
  - Exclude range of IP addresses from pool: ip dhcp excluded-address ⟨Start IP address⟩ ⟨End IP address⟩
- Show interfaces:

- enable
- show interfaces
- show ip interface [brief]

▸ Set up a static route:

- enable>configure terminal
- interface ⟨Interface⟩
- ip route ⟨DST Network⟩ ⟨DST Subnetmask⟩ ⟨Gateway⟩, where ⟨Gateway⟩ refers to the IP address of the interface of the router used to access the ⟨DST Network⟩, e.g. ip route 192.168.12.0 255.255.255.128 11.12.0.1. Note that for on-link routes the Gateway is given by the interface, e.g. ip route 192.168.12.0 255.255.255.128 GigabitEthernet 0/0.

▸ Show routes:

- enable
- show ip route

▸ Remove a static route:

- enable>configure terminal
- no ip route ⟨DST Network⟩ ⟨DST Subnetmask⟩ ⟨Gateway⟩, e.g. ip route 192.168.12.0 255.255.255.128 11.12.0.1

▸ Clear routing table (does not remove static routes):

- enable
- clear ip route *

8.7 Problem 10: Media Access Control

▸ Q: Since a layer 2 frame contains only MAC addresses, how do PCs know which MAC addresses another PC/tablet has?
A: Initially, the PCs/tablets do not know the physical MAC addresses of the other devices, but they infer them by broadcasting an ARP request which uses a logical IP address to identify the devices. Afterwards, the MAC address is kept in an ARP table in case it is needed for future communication. This ARP table can be shown with arp -a and emptied using arp -d.

▸ Q: Which layer 2 information does a switch save and how can you observe this information?
A: A switch records the source MAC addresses of devices sending (Ethernet) frames via them. They can be inferred from the CLI using enable>show mac-address-table.

▸ Q: Why does ICMP communication (ping) between the tablet and the PCs work with the tablet being connected by Wi-Fi while the PCs use an Ethernet cable?
A: Ethernet and Wi-Fi are compatible layer 2 protocols. While an Ethernet copper cable guides the information as an AC current, electromagnetic waves transport the information through the air. Since electromagnetic waves emerges as radiation of an AC current in an antenna and electromagnetic waves can induce an AC current in a receiver, both can be transformed into another.

8.8 Problem 11: MAC address handling

▸ Q: What happens in the layer 2 PDU during communication (ping) between PC0 and PC3?
A: Without a freshly installed network, the ping is interrupted by several ARP requests:

- ARP request by PC0: Get MAC address of Router0
- ping by PC0 to PC3:
  - ARP request by Router0: Get MAC address of Router1
  - ↪ Timeout of ping
- ping by PC0 to PC3:
  - ARP request by Router1: Get MAC address of PC3
  - ↪ Timeout of ping
- ping by PC0 to PC3

After this initial run, the pings proceed uninterrupted. During this second run, the following changes happen on layer 2:

- PC0: SRC MAC = MAC_{PC0}, DST MAC = $\text{MAC}_{\text{R0,I0/0}}$
- R0: SRC MAC = $\text{MAC}_{\text{R0,I0/1}}$, DST MAC = $\text{MAC}_{\text{R1,I0/1}}$
- R1: SRC MAC = $\text{MAC}_{\text{R1,I0/0}}$, DST MAC = MAC_{PC4}
- R1: SRC MAC = MAC_{PC4}, DST MAC = $\text{MAC}_{\text{R1,I0/0}}$
- R0: SRC MAC = $\text{MAC}_{\text{R1,I0/1}}$, DST MAC = $\text{MAC}_{\text{R0,I0/1}}$
- PC0: SRC MAC = $\text{MAC}_{\text{R0,I0/0}}$, DST MAC = MAC_{PC0}

Essentially once the (Ethernet) frame has reached a device, the DST MAC address becomes the new SRC MAC address and the MAC address of the device next in line is entered as the DST MAC address.

Q: Do routers record information on the MAC address? If yes, how can one access them?
A: Yes, routers record information on MAC addresses when using ARP to infer the MAC address of a device when it is given that devices IP address. The ARP table can be displayed with enable>show arp and shows the devices MAC address and IP address as well as the routers interface connected to the network of the device. To clear the ARP table on routers, use clear arp-cache and on PCs use arp -d. To prevent an ensuing ARP request to repopulate the ARP table, the interface has to be shutdown.

Another command to show MAC address information is enable>show mac-address-table. On a router without switch module, the resulting MAC address table is empty. On a switch or a router with switch module, this command displays the device MAC addresses and the switch ports these devices are connected to. To clear the MAC address table, use clear mac-address-table.

In summary, the ARP table is the result of layer 3 activity (ARP requests), while the MAC address table is the result of layer 2 activity (layer 2 interfaces receive frame and discover SRC MAC address).

8.9 Problem 12: ARP

Q: What happens during a ping from PC0 to PC1? In which order are packages sent? Why is this the case?
A: First PC0 sends out an ARP request as broadcast (DST MAC: FFFF.FFFF.FFFF) to get the MAC address belonging to the IP address. Once the ARP request reaches PC1, it replies with its MAC address. PC0 then uses MAC and IP address to ping PC1.

The ARP request comes before the ICMP message since the MAC address identifies a device in a local network. In terms of the OSI model, the MAC address is required since after the bits/bytes are reassembled into an (Ethernet) frame, its header with the MAC address is examined in layer 2 to determine whether it is meant for the device looking into it. The IP address encapsulated within the (Ethernet) frame is only accessible once the MAC address has been accepted and the (Ethernet) frame is unwrapped showing the IP packet with the IP address in the header. Without the ARP request and with an arbitrary MAC address, the (Ethernet) frame would be rejected.

Without MAC address, a broadcast is necessary. If instead of broadcasting the ARP request every message would be sent as a broadcast, the entire network would end up suffering from congestion. While an ARP request is small, messages are large and can be transported as several (Ethernet) frames. It thus makes more sense to request the MAC address once for a succession of messages.

Q: During a ping from PC0 to PC3, which packages are sent besides the ICMP ones? Why is this the case? How many ARP requests are there?
A: During a ping from PC0 to PC3, the following happens (see section 8.8):

- ARP request by PC0: Get MAC address of Router0
- ping by PC0 to PC3:
  - ARP request by Router0: Get MAC address of Router1
  - ↪ Timeout of ping
- ping by PC0 to PC3:
  - ARP request by Router1: Get MAC address of PC
  - ↪ Timeout of ping
- ping by PC0 to PC3

Besides the ICMP message (ping), three ARP requests are sent to determine the MAC addresses of the default gateway, the router leading to the network containing the TGT IP and the device having the TGT IP. These MAC addresses are necessary to move the ICMP message to its destination.

- Q: How does an ARP request spread in the network?
  A: Since an ARP request is sent as a broadcast (DST MAC: FFFF.FFFF.FFFF), every switch sends it off to each device connected to the switch. Since router operate on layer 3 and require IP addresses, they do not propagate ARP requests. In the end, only the device with the TGT IP in the ARP request replies with its (then SRC) MAC address.
- Q: How long does the ARP cache save an entry?
  A: On Ubuntu or ArchLinux, the file /proc/sys/net/ipv4/neigh/default/gc_stale_time sets the time when the next entry in the ARP table is removed to 1 min. Every time the next entry to be removed is used, the timer is reset to 1 min. On Windows, the time lies between 15 and 45 s and can be inferred by running
  - Command: netsh interface ipv4 show interfaces →⟨Interface ID⟩
  - Command: netsh interface ipv4 show interface ⟨Interface ID⟩ →Reachable Time
- Q: Why does the router have an ARP table? How can one display it? How long does the router save the entries?
  The router has an ARP table to determine the MAC addresses of other devices in its network and associate them with IP addresses. This is necessary to be able to send packages to other devices since the MAC address is the first identifier to be checked. To display the ARP table on a router, one uses the command show ip arp. A Cisco router flushes an entry from its ARP table after 4 h during which no packet were sent to/received from a pair of MAC and IP addresses.
  Note that while switches do not perform ARP requests, they have a table mapping switch ports to MAC addresses. After 5 min of not receiving (Ethernet) frames from a MAC address, it is removed. Each time, (Ethernet) frames from a MAC address are received, the timer is updated. In case there are multiple ports with the same MAC address, the port with highest timer is used.

8.10 Problem 13: DHCP relay

- Q: What did you configure on the router to allow the DHCP DISCOVER messages reach the DHCP server?
  A: The need to configure the router comes with the DHCP DISCOVER message being a broadcast (DST MAC: FFFF.FFFF.FFFF, DST IP: 255.255.255.255), which is usually not forwarded by a router. To enable forwarding UDP broadcasts, the ⟨Interface⟩ facing the network from which such a broadcast can come has to be configured using
  - enable/configure terminal/interface ⟨Interface⟩
  - ip helper-address ⟨IP address⟩

  where ⟨IP address⟩ is that of the DHCP server in another network connected to the router.
- Since the procedure to obtain an IP address from DHCP in this setup is involved, it is described in the following:
  - First DHCP request:
    - DHCPDISCOVER message by PC0: The DHCP message (SRC MAC: MAC_{PC0}, CLIENT MAC: MAC_{PC0}) is broadcast. Due to its configuration, R0 registers the message.
    - ARP request by R0 from DHCP server: Get the MAC address of the DHCP server using the helper IP address on R0

    ↳ R0 gets MAC address of DHCP server, DHCPDISCOVER times out

    After this first DHCP request has timed out, PC0 claims the IP address 169.254.151.129/16 lying in the address range 169.254.0.0/16, which are used by devices without IP address that should have received one (e.g. from a DHCP server). PC0 sends out an ARP request to check if another device in the network already has this IP address and keeps it, once no answer returns.
  - Second DHCP request:
    - DHCPDISCOVER message by PC0: The DHCP message (SRC MAC: MAC_{PC0}, CLIENT MAC: MAC_{PC0}) is broadcast. R0 passes the message to the DHCP server.
    - ICMP message by DHCP server to R0: Check if offered IP address is free
    - ARP request by R0 in network of PC0 with offered IP address: Identify network of PC0 by using network ID of offered IP. Check if offered IP address is free.

    ↳ No reply
    - ICMP message by DHCP server: Check if offered IP address is free
    - R0 rejects ICMP message

- DHCP server sends DHCPOFFER (SRC MAC: MAC_{R0}, CLIENT MAC: MAC_{PC0}) to R0
- R0 passes DHCPOFFER as broadcast (DST MAC: FFFF.FFFF.FFFF, DST IP: 255.255.255.255) into network of PC0
- PC0 receives DHCPOFFER due to CLIENT MAC: MAC_{PC0}
- DHCPREQUEST by PC0 from DHCP server
- DHCPACK by DHCP server to PC0
- ICMP message by DHCP server: Check if offered IP address is free
- R0 rejects ICMP message
- PC0 receives DHCPACK
- ARP request by PC0: Check if offered IP address is free

For the other two devices only a single DHCP request is required each, since the router obtained the MAC address of the DHCP server previously. In case the IP address offered by the DHCP server is already assigned to a device, R0 forwards the DHCPOFFER to this device instead of the origin of the DHCP request. Since it already has an IP address and did not send a DHCPDISCOVER message, it does not reply with a DHCPREQUEST.

- To prove that the IP address is dynamical, ipconfig /release is executed and does not return "Port is not using DHCP.".

8.11 Problem 14: Switches and VLANs

- Q: What are VLANs?
  A: While the physical layer contains the physical cabling, virtual local area networks (VLANs) represent a type of logical cabling. By splitting switch ports into groups which only connect among themselves, the flow of data can be controlled on a software level.
- Q: What is the purpose of VLANs?
  A: The original motivation for VLANs was that while Ethernet has a large bandwidth, it is not scalable. To connect Ethernet networks via switches in a fault-tolerant fashion requires redundant paths that imply loops. Broadcasts in a network with loops lead to Broadcast storms that block all normal communication. To prevent loops, several ports of switches have to be blocked to achieve a spanning tree configuration with only one active path from each source to each destination within the network. This however turns centrally located switches into bottlenecks. Directly linking Ethernet networks is thus not a good idea and using routers is slow and expensive.

  Considering several Ethernet networks, each in a spanning tree configuration, and running them disconnected from one another in parallel keeps the loops away while increasing the bandwidth along paths covered by multiple different Ethernet networks. Starting from the other side with one Ethernet network and separating it into parts running in parallel, effectively separates the broadcasting domain into several smaller ones.

  In summary, VLANs allow to scale up Ethernet networks in size, enhance network security by separating data streams and manage traffic by allowing for parallel paths without suffering due to broadcasts.
- Q: What changes about an (Ethernet) frame when using VLANs?
  A: To implement VLANs, an additional 12 bit long VID (VLAN identifier) field is added to each (Ethernet) frame to allow ports of switches to block them if they do not have the right tag. Note that VLANs with different tag are connected by routers which block broadcasts but allow unicast (Ethernet) frames to pass.
- Q: What is a trunk port?
  A: In the context of VLANs, the ports of switches are divided into two types, access/untagged ports and trunk/tagged ports.

  Devices attached to an access port are not aware of the logical decomposition into VLANs. Any (Ethernet) frames sent or received via access ports do not have a specific tag. Access ports connect devices only to a single VLAN. By default, every port of a switch is a an access port.

  Trunk ports only connect devices aware of the logical decomposition into VLANs (e.g. switches, routers). They add a tag to (Ethernet) frames before they are passing them on. This is necessary since connections between trunk ports transfer (Ethernet) frames from different VLANs and the receiving device requires the tag to associate the (Ethernet) frames to the correct VLAN. No switch port is by default a trunk port.

  Each switch has a VLAN specified as its native VLAN. If an end device or the access port of a switch is

(accidentally) connected to the trunk port of another switch, the (Ethernet) frame received is considered untagged and associated with the native VLAN of the receiving switch. It is essentially the default VLAN for untagged (Ethernet) frames received over a trunk port. This way the tags of (Ethernet) frames can accidentally change which should not be used systematically and is the sign of an erroneous network.

If a tagged (Ethernet) frame coming from a trunk port is received by an end device or the access port of another switch, it is usually dropped.

To configure the switches, the following commands will be used:

- Create a VLAN ⟨V-ID⟩ (VLAN ID) on a switch:
  - enable>configure terminal
  - vlan ⟨V-ID⟩
- Assign an interface FastEthernet 0/⟨P-ID⟩ (port ID) to be an access port (connection device-switch) for VLAN ⟨V-ID⟩, i.e. make the device connected to this interface part ot VLAN ⟨V-ID⟩:
  - enable>configure terminal/interface FastEthernet 0/⟨P-ID⟩
  - switchport mode access
  - switchport access vlan ⟨V-ID⟩
-
- Assign an interface FastEthernet 0/⟨P-ID⟩ to be a trunk port (connection switch-switch or switch-router) for VLAN ⟨V-ID⟩:
  - enable>configure terminal/interface FastEthernet 0/⟨P-ID⟩
  - switchport mode trunk
  - switchport trunk allowed vlan ⟨V-ID⟩

  Note that a trunk port allows multiple VLANs, hence ⟨V-ID⟩ can be a range (e.g. 2–4) or a comma separated list (e.g. 10,25,35).
- Show the VLANs and the interfaces used as access ports:
  - enable
  - show vlan
- Show the interfaces used as trunk ports with the VLANs they transmit:
  - enable
  - show interfaces trunk
- Give a switch an ⟨IP address⟩ within VLAN ⟨V-ID⟩:
  - enable>configure terminal>interface vlan ⟨V-ID⟩
  - ip address ⟨IP address⟩ ⟨Subnetmask⟩
- Show IP address of switch in VLAN ⟨V-ID⟩:
  - enable
  - show interfaces vlan ⟨V-ID⟩
- Configure switch for ssh:
  - Give the switch in a VLAN of choice an IP address
  - enable>configure terminal
  - Set host name, domain name:
    - hostname ⟨host name⟩
    - ip domain-name ⟨domain name⟩
  - Configure primary terminal line:
    - line console 0
    - password ⟨password⟩
    - logging synchronous →synchronized message output
    - login local →local password checking, exit
    - enable secret ⟨password⟩
  - Generate ssh key: crypto key generate rsa general-keys modules 1024
  - Configure virtual terminal:
    - line vty 0 4
    - transport input ssh →use ssh as transport protocol

    - login local →local password checking
    - password ⟨password⟩, exit
  - Create user, password: username ⟨user name⟩ password ⟨password⟩

- Set up an access list (ACL):
  - enable>configure terminal
  - ip access-list standard ⟨number⟩
  - Blacklist:
    - deny host ⟨IP address⟩
    - deny ⟨IP address⟩ ⟨Subnetmask⟩
    - permit any, exit

  ↳ permit all but the specified IP addresses
  - Whitelist:
    - permit host ⟨IP address⟩
    - permit ⟨IP address⟩ ⟨Subnetmask⟩
    - deny any, exit

  ↳ deny all but the specified IP addresses

To set up the VLANs, the switches are configured as follows:

- Switch0:
  - Create VLANs 10, 25 and 35
  - Assign interface FastEthernet 0/1 as access port for VLAN 10
  - Assign interfaces FastEthernet 0/23 and 0/24 as trunk ports for VLANs 10, 25 and 35
  - Give the switch the IP address 10.12.10.200 within VLAN 10
  - SSH access:
    - host name: switch00
    - domain name: grimshaw.com
    - password, secret, password: stitches, stitches, stitches
    - username, password: jan, stitches00

- Switch1:
  - Create VLANs 10, 25 and 35
  - Assign interfaces FastEthernet 0/1 and 0/2 as access ports for VLAN 35
  - Assign interfaces FastEthernet 0/3 as access port for VLAN 25
  - Assign interface FastEthernet 0/23 as trunk port for VLANs 10, 25 and 35
  - Give the switch the IP address 10.12.10.25 within VLAN 10
  - SSH access:
    - host name: switch11
    - domain name: grimshaw.com
    - password, secret, password: stitches, stitches, stitches
    - username, password: jan, stitches11

- Switch2:
  - Create VLANs 10, 25 and 35
  - Assign interfaces FastEthernet 0/1 and 0/2 as access ports for VLAN 25
  - Assign interfaces FastEthernet 0/3 as access port for VLAN 35
  - Assign interface FastEthernet 0/23 as trunk port for VLANs 10, 25 and 35
  - Give the switch the IP address 10.12.10.100 within VLAN 10
  - SSH access:
    - host name: switch22
    - domain name: grimshaw.com
    - password, secret, password: stitches, stitches, stitches
    - username, password: jan, stitches22

To restrict the access via ssh, access lists can be set up as described above.

When pinging PCs, the (Ethernet) frames between switches connected by trunk ports display an Ethernet 802.1q header in place of the Ethernetll header. This header contains additionally the tag protocol identifier (TPID) and tag control information (TCI) fields. While the 16 bit long TPID field merely contains the value

0x8100 for VLANs, the 16 bit long TCI field has more structure: 3 bit contain the priority code point (PCP) which specifies the frame priority level, the next 1 bit contains the drop eligible indicator (DEI) which indicates frames that can be dropped in presence of a congestion and the last 12 bit denote the actual VLAN ID.

8.12 Problem 15: Inter-VLAN routing and everything combined

8.12.1 Dedicated DHCP server

- Q: Which type of packet (Unicast, multicast or broadcast) are sent when requesting and receiving a dynamic IP address and how are these four packets called?
  A: The four packets echanged when requesting and receiving a dynamic IP address are sent as broadcast and called DISCOVER, OFFER, REQUEST and ACKNOWLEDGEMENT.

8.12.2 DHCP server on a router

- Q: How is the DHCP pool of the router configured?
  A: Following the description in section 8.6, the DHCP pool is configured using the following commands:
  - enable>configure terminal
  - ip dhcp pool pool10
  - network 10.0.24.0 255.255.255.0
  - default-route 10.0.24.254
  - dns-server 0.0.0.0
  - ip dhcp excluded-address 10.0.24.51 10.0.24.255

8.12.3 Inter-VLAN routing

- Q: Why is a layer 3 device necessary to enable communication between different subnets coinciding with different VLANs?
  A: One purpose of VLANs is to separate broadcast domains. This means that an ARP request broadcast by a device in one VLAN to infer the IP address of another device will not reach this device if it is located in another VLAN. To allow a broadcast to reach a device in a specific VLAN and a different subnet requires a router providing a routing table to that subnet.

  Note that an (Ethernet) frame sent from an access port to the trunk port of another switch will be interpreted as belonging to the native VLAN of the receiving switch. The (Ethernet) frame will thus reach its destination if the end device is part of the switches native VLAN. However the response to this (Ethernet) frame may not reach its destination. Furthermore, if the device is not part of the native VLAN, the (Ethernet) frame fails to reach its destination.
- Q: What is a VLAN tag?
  A: A VLAN tag is used on connections between trunk ports to distinguish (Ethernet) frames from different VLANs. Such tags allow to replace the three connections between the switch and the router physically separating the VLANs by a single one while tagging the (Ethernet) frames. The setup using a single connection and tagged (Ethernet) frames is called “router-on-a-stick”.

8.12.4 Routing

To connect all previously discussed separate networks, the routers are configured with the following routing tables:

- Router 0:

Destination	Gateway	Interface
4.4.4.0/29	on-link	FastEthernet 3/0
6.6.6.0/29	on-link	FastEthernet 4/0
172.16.90.0/24	on-link	FastEthernet 0/0

172.16.10.0/24	on-link	FastEthernet 1/0
172.16.50.0/24	on-link	FastEthernet 2/0
10.0.24.0/24	6.6.6.2	6.6.6.1
192.168.16.0/24	4.4.4.1	4.4.4.2

- Router 1:

Destination	Gateway	Interface
5.5.5.0/29	on-link	GigabitEthernet 0/2
6.6.6.0/29	on-link	GigabitEthernet 0/1
10.0.24.0/24	on-link	GigabitEthernet 0/0
172.16.10.0/24	6.6.6.1	6.6.6.2
172.16.50.0/24	6.6.6.1	6.6.6.2
172.16.90.0/24	6.6.6.1	6.6.6.2
192.168.16.0/24	5.5.5.2	5.5.5.1

- Router 2:

Destination	Gateway	Interface
4.4.4.0/29	on-link	GigabitEthernet 0/1
5.5.5.0/29	on-link	GigabitEthernet 0/2
192.168.16.0/24	on-link	GigabitEthernet 0/0
10.0.24.0/24	5.5.5.1	5.5.5.2
172.16.10.0/24	4.4.4.2	4.4.4.1
172.16.50.0/24	4.4.4.2	4.4.4.1
172.16.90.0/24	4.4.4.2	4.4.4.1

Pinging every device from every other device in the entire network verifies this configuration.

8 12 5 Router on a stick

To enable inter-VLAN routing by means of router on a stick, the involved switch and router have to be configured:

- Switch with interface ID ⟨S-ID⟩:
  - enable>configure terminal>interface FastEthernet 0/⟨S-ID⟩
  - switchport mode trunk

  Note that in case only specific VLANs are supposed to interconnect at the router, the allowed VLANs have to be specified according to section 8.11.
- Router with interface ID ⟨R-ID⟩:
  - enable>configure terminal
  - Create subinterface with ID ⟨R-IDsub⟩: interface FastEthernet ⟨R-ID⟩/0.⟨R-IDsub⟩
  - Assign subinterface to VLAN with ID ⟨V-ID⟩: encapsulation dot1q ⟨V-ID1⟩
  - Give IP address to subinterface: ip address ⟨IP address⟩, exit
  - interface FastEthernet ⟨R-ID⟩
  - no shutdown, exit

  For each VLAN ⟨V-ID⟩ to interroute between, a subinterface ⟨R-IDsub⟩ assigned to the VLAN with an IP address ⟨IP address⟩ has to be created. Note that it is a good idea to choose for the subinterface ID the same value as the VLAN ID, i.e. ⟨R-IDsub⟩ = ⟨V-ID⟩.

For the present network, the switch and router are configured as follows:

- Switch: The interface FastEthernet 0/24 is configured as trunk port.

- Router: The interface FastEthernet 0/0 is used to connect to the switch. On it the following subinterfaces are configured:

Subinterface	VLAN	IP address
0.4	4	172.16.90.254
0.15	15	172.16.10.254
0.22	22	172.16.50.254

Pinging every device from every other device in the entire network verifies this configuration. The same is done in the complete network with three routers.

Wireshark:

- record frames, display recordings understandable
- search and filter
- identify streams/sessions, can extract files passed in a session
- statistics, graphs on traffic →identify origin of congestion
- Setup: network →capture filter →capture engine →dissector, display filter, display →dump
- capture filter: filter out certain traffic (exclude certain traffic from being recorded →less data collected, save memory), makes sense if searching for certain things, makes less sense if origin of error in network unclear
- capture engine: collects data, puts it into dump
- dissector process collected data into different parts (e.g. IP, TCP, UDP) to interpret and display headers of packets
- display filter: only show certain things
- dump: save collected data on hard drive
- promiscuous mode:
  - NIC ignores frames not intended for device (MAC address)
  - promiscuous mode: NIC collects all incoming frames independent of destination MAC address
  - Wireshark requires sudo/admin rights
  -
  -
- Settings/Preferences:
  - Name Resolution →Resolve MAC addresses: obtain manufacturer name from first bytes of MAC address →identify devices
  - Protocol: setup dissector for different protocols
  - TCP: deactivate "Allow subdissector to reassemble TCP streams"→otherwise: loss of packets in display (deactivate to show all)
  - Calculate conversation timestamps: computes time relative to first packet received →get length of conversation, also time estimates for process length →detect congestion, loss of data (multiple sends), switch does something strange
- Where to use Wireshark for recording
  - does the traffic I want to observe even pass this device?
  - connection problems to server →run Wireshark at server
  - one client causes /has problems →run Wireshark at client
  - assume man-in-the-middle attack →run Wireshark at outgoing router →if all outgoing packets come from same MAC address: attacker in network mimics to be default gateway and passes packets from clients (all packets show same address for default gateway)
  - firewalls might block certain traffic →run Wireshark before or after firewall
  - run Wireshark before or after NAT
  - run Wireshark as close to assumed origin of problem as possible
  - get authentic Wireshark recording (avoid feedback from Wireshark onto network)
  - issue: Wireshark running on problematic device affects that device (RAM fills with captured traffic, CPU runs process)
  - question: Does Wireshark alter situation? run Wireshark somewhere it is not obvious
- Start recording

- set promiscuous mode under Options>Input for interface
- set capture filter under Options (only use if necessary, i.e. running out of memory →better: use display filter)
- Options>Output:
  - Specify file in which to save recorded traffic (uses temporary file if empty) →if longer observation: file gets large →performance suffers →Create a new file automatically... (after size of file)
  - Multiple small files: better performance, more files to search through
  - Use a ring buffer: Cyclically overwrite files →only few files, good performance, last kept files can be examined

↳ What do I want to see?

▸
▸
▸

▸ Wireshark at a switch:

- Problem with switches: first ARP request may be send as broadcast, but after a switch has assigned its ports to MAC addresses, a machine running Wireshark is excluded from the communications between other devices via the switch
- Network TAP: device with at least three ports (A, B, monitor) between switch and one of the communicating devices →copy of communication is passed in realtime to monitor port

↳ Network TAP does not cause feedback for rest of network

- SPAN (switched port analyzer)/Mirror ports: on a managed switch one can configure mirror ports which copies the traffic running to/from (depends what can be configured) one (source) port to another (destination) port (monitoring)

– issue: switch busy with making copies (buffering, CPU) which may have a feedback on the network and could affect the troubleshooting
– multiple ports can be mirrored onto another port, but the single (destination) port has be capable of passing all the traffic running through all mirrored ports →Is this even possible?

- HUBs: use a HUB instead of a TAP, i.e. put a HUB between switch and one of the communicating devices and connect the monitoring device and the communicating device to the HUB
- issues: collisions since monitoring computer not silent →modifies network behavior, may not be capable of full duplex →switch may change port to half duplex, network behavior heavily affected

▸ Display filter:

- What is a display filter: Comes after the dissectors and filter the captured traffic
- Working with display filters:
  - case sensitive
  - setup: <protocol>.<element>.<subelement> <operator> <value>
  - protocol: TCP, UDP, IP, DNS
  - element: port number, IP address
  - operator: == (eq), != (neq), < (lt), > (gt), <= (le), >= (ge), and (&&), or (||), not (!)
  - simple protocol filters: ip, tcp, arp, icmp
  - with element: ip.addr == 10.0.1.11, tcp.port == 80
  - with subelement: tcp.flags.fin == 1
  - brackets to group filters (follow rules of propositional logic)
  - keyword contains: tcp contains “google”→show messages containing keywords
- Backgroundcolors of filter field:
  - red: syntax error
  - green: syntax correct
- Often used filter options:
  - tcp.flags.syn == 1: detect DOS attack by checking how many tcp messages with syn flag set arrive
  - right click>Follow>TCP stream: shows all segments belonging to a specific TCP connection

OSI model demystified:

▸ purpose of OSI model from technologist perspective: helps to think about networking problem, way to localize at which stage (layer) the problem occurs (→place to start troubleshooting from) and determine a way to fix the problem

- purpose of OSI model from programmer perspective: figure out what system (layers, protocols) they have to build the software for
- practical usage of OSI model: troubleshoot, identify problem
- Transport layer: decides how much information should be send at one time (from client, from server), windowing: process where computers send information back and forth, decides how large a block of information to send should be, decides how long the computer should wait before it receives an acknowledgement that information was sent or received
- IT people normally don't play with transport level very much (either works or doesn't)
- layer 7 problems: firefox, outlook corrupted, misconfiguration
- layer 6 problems: device drivers messed up, something within operating system does not allow user to access Internet (maybe not right security protocols)
- layer 5 problems: IT people sometimes deal with this (not often), when administering a website there may be settings in the webserver causing problems with the session layer (PHP config files, apache config files) →sessions may not be able to connect
- layer 4 problems: was important in the time of dial-up modems
- layer 3 problems: problems with routers (router dies), IP addresses, default gateways, subnet masks, DNS (entering wrong IP address not leading to the intended location)
- layer 2 problems: problems with switches (switch not working properly, information not send properly, slow network due to switches)
- layer 1 problems: no patch cable, network cable not connected, cable gets cut, cable miswired
- claim: 95% of networking problems are layer 1 problems (somebody unplugged the cable)

Bibliography

[BBC19] BBC, *Christchurch shootings: 49 dead in New Zealand mosque attacks*, https://www.bbc.com/news/world-asia-47578798, Accessed on 02.10.2023, 5:35 pm, 2019.

[Car21] Carson, *TCP 3-Way Handshake and How it Works*, https://cabulous.medium.com/tcp-3-way-handshake-and-how-it-works-8c5f8d6ea11b, Accessed on 20.10.2023, 11:30 am, 2021.

[CH23] P. Christiansen and C. Haynes, *How Does the Internet Work?*, https://www.highspeedinternet.com/resources/how-the-internet-works, Accessed on 02.10.2023, 5:35 pm, 2023.

[Cis23] Cisco, *Cisco Network Topology Icons 3015*, https://vecta.io/symbols/240/cisco-network-topology-icons-3015, Accessed on 22.10.2023, 9:30 am, 2023.

[Cra19] A. Crawford, *Instagram eating disorder content 'out of control' (BBC)*, https://www.bbc.com/news/uk-47637377, Accessed on 02.10.2023, 5:35 pm, 2019.

[Dav17] DavidPostill, *Are there well known HTTP-only sites?*, https://superuser.com/questions/1213116/are-there-well-known-http-only-sites, Accessed on 23.10.2023, 8:30 am, 2017.

[Dej20] V. Dejwakh, *How the Internet Works, Part I - The Internet Infrastructure*, https://vahid.blog/post/2020-12-15-how-the-internet-works-part-i-infrastructure/, Accessed on 02.10.2023, 5:35 pm, 2020.

[Fou23a] W. Foundation, *Client-Server-Architektur*, https://en.wikipedia.org/wiki/Client-server_model, Accessed on 04.10.2023, 2:30 pm, 2023.

[Fou23b] W. Foundation, *History of the Internet*, https://en.wikipedia.org/wiki/History_of_the_Internet, Accessed on 04.10.2023, 8:10 am, 2023.

[Fou23c] W. Foundation, *Networking hardware*, https://en.wikipedia.org/wiki/Networking_hardware, Accessed on 04.10.2023, 8:10 am, 2023.

[Hat18] R. Hatzipanagos, *How online hate turns into real-life violence (Washington Post)*, https://www.washingtonpost.com/nation/2018/11/30/how-online-hate-speech-is-fueling-real-life-violence/, Accessed on 02.10.2023, 5:35 pm, 2018.

[IPC18] IPCisco.com, *Router DHCP Configuration with Packet Tracer*, https://ipcisco.com/lesson/router-dhcp-configuration-with-packet-tracer-ccna/, Accessed on 24.10.2023, 10:00 am, 2018.

[Kom23a] E. Kompendium, *Client-Server-Architektur*, https://www.elektronik-kompendium.de/sites/net/2101151.htm, Accessed on 04.10.2023, 2:30 pm, 2023.

[Kom23b] E. Kompendium, *Netzwerk-Komponenten*, https://www.elektronik-kompendium.de/sites/net/0505221.htm, Accessed on 04.10.2023, 8:10 am, 2023.

[Kom23c] E. Kompendium, *What Does a Network Engineer Do? A Comprehensive Guide for 2023*, https://emeritus.org/blog/technology-network-engineer-do/, Accessed on 04.10.2023, 8:10 am, 2023.

[Koz17] C. Kozierok, *The TCP/IP Guide: A TCP/IP Reference You Can Understand!*, http://www.tcpipguide.com/index.htm, Accessed on 22.10.2023, 9:30 am, 2017.

[Sam20] E. Samuels, *How misinformation on WhatsApp led to a mob killing in India (Washington Post)*, https://www.washingtonpost.com/politics/2020/02/21/how-misinformation-whatsapp-led-deathly-mob-lynching-india/, Accessed on 02.10.2023, 5:35 pm, 2020.

[str11] stretch, *TCP Flags: PSH and URG*, https://packetlife.net/blog/2011/mar/2/tcp-flags-psh-and-urg/, Accessed on 20.10.2023, 11:30 am, 2011.

[Str21a] J. Strickland, *How Does the Internet Work?*, https://computer.howstuffworks.com/internet/basics/internet.htm, Accessed on 02.10.2023, 5:35 pm, 2021.

[Str21b] J. Strickland, *What is a packet?*, https://computer.howstuffworks.com/question525.htm, Accessed on 02.10.2023, 5:35 pm, 2021.

[Str23] J. Strickland, *How did the Internet start?*, https://computer.howstuffworks.com/internet/basics/internet-start.htm, Accessed on 02.10.2023, 5:35 pm, 2023.

[TP23] J. Tyson and C. Pollette, *How Internet Infrastructure Works*, https://computer.howstuffworks.com/internet/basics/internet-infrastructure.htm, Accessed on 02.10.2023, 5:35 pm, 2023.