# Hoofdstuk 1: Explore the Network

1.2.1 : Network Components

Services: Netwerk applicaties (email hosting, webhosting).

End Devices: Computer (met NIC), printer, smartphone IP-telefoon… (Source of destination van bericht).

Intermediary Devices: Router, switch, firewall 🡪 Kiezen welke weg berichten moeten nemen.

Een router geeft toegang tot het internet (het kent de weg op het internet)

Switch: Vindt de weg op een lokaal netwerk.

Physical Port: Een connector op een netwerktoestel waaraan de media (kabel) geconnecteerd is met een end device of een ander netwerktoestel.

Interface: Een poort op een netwerktoestel dat verbonden is aan een individueel netwerk. Omdat routers netwerken verbinden, worden de poorten op een router interfaces genoemd.

1.2.2: LANs and WANs

**LAN**: Klein geografisch gebied, huis of klein bedrijf. **WLAN:** Wireless. Een LAN dat radiogolven gebruikt.

**PAN:** Personal Area Network, klein netwerk, met muis, printer, smartphone, laptop… (Bluetooth).

**MAN:** Metropolitan Area Network. Overspant een grote campus of stad, het bestaat uit meerdere gebouwen, die tussen elkaar verbonden zijn. (Connecties van LANs).

**WAN:** Enorm groot. Verbind meerdere netwerken die geografisch ver gelegen zijn. (Vb. Internet).

**SAN:** Storage Area Network (Voor File-Servers).

Het Internet is een wereldwijde collectie van interconnecteerde netwerken. (LANs en WANs).

Intranet: Gemaakt om binnen een organisatie gebruikt te worden. Het is een privé collectie van LANs en WANs dat tot een organisatie behoort.

Extranet: Geeft toegang tot de data van een organisatie aan mensen buiten de organisatie.

1.3.1: Converged Networks

Vroeger: verschillende netwerken voor video, telefoon en data. Nu: Convergerende netwerk: Video, telefoon en data kunnen dezelfde netwerkinfrastructuur gebruiken, met dezelfde regels.

1.3.2: Reliable Network

4 karakteristieken voor een betrouwbaar netwerk: Fault Tolerance, Scalability, Quality of Service (QoS), Security

Fouttolerantie: Redundancy, meerdere wegen tussen bron en bestemming op een netwerk. Als de ene weg faalt, bestaat een andere weg naar de bestemming.

Scalability: Makkelijk uit te breiden voor nieuwe gebruikers en applicaties zonder impact op prestaties.

QoS: Netwerk dat niet te traag is (buffering van videos), waar geen opstopping is. (Meer bandwith dan mogelijk vragen). 🡪 Streaming krijgt hogere prioriteit dan webpagina’s.

Beveiliging: Data wordt niet veranderd, data blijft confidentieel, data blijft beschikbaar.

1.4.1: Network Trends

Bring Your Own Device: BYOD. Leerling kan eigen laptop, smartphone gebruiken op school.

Cloud Computing: Mogelijk door datacenters. 4 Soorten clouds: Publieke, privé, custom en hybride.

Powerline Networking: Bestaande elektriciteitskabels gebruiken voor data.

1.4.3: Network Security

Netwerk beschermen: Met antivirus, antispyware, firewall.

# Network Concepts: Hoofdstuk 2

2.1.1: Cisco IOS

-Hardware: The physical part of a computer including underlying electronics.

-Kernel: Communicates between the hardware and software of a computer and manages how hardware resources are used to meet software requirements.

-Shell: The user interface that allows users to request specific tasks from the computer. (These requests can be made either through the CLI or GUI interfaces.)

2.1.2: Cisco IOS-access

Hoe kunnen we een switch configureren?

**Console** – A physical port that provides out-of-band access to a network device. Out-of-band access refers to access via a dedicated channel that is used for device maintenance purposes only.

**Secure Shell (SSH)** – A method for remotely establishing a secure CLI connection through a virtual interface, over a network. SSH connections require active networking services on the device including an active interface configured with an address.

**Telnet** - Telnet is an insecure method of remotely establishing a CLI session through a virtual interface, over a network. User authentication, passwords, and commands are sent over the network in plaintext. (Geen encryptie)

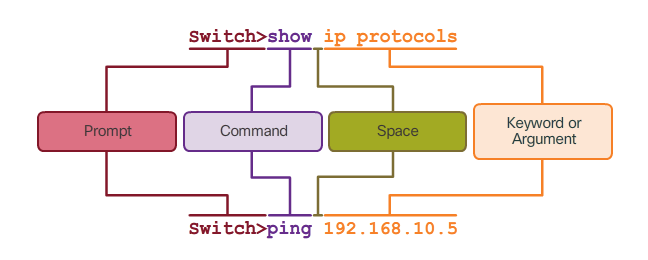
- Ook **AUX**, bijvoorbeeld over telefoonconnectie, als je geen toegang hebt via het internet.

2.1.3: Navigate the IOS

Er bestaan verschillende commando modes. De **User Exec Mode**> veel beperkingen, enkele basiscommando’s, en de **Privileged Exec Mode**#, waarmee we kunnen configureren. Om naar Privileged Exec mode te gaan typen we Switch> **Enable**. We kunnen nog hoger gaan, in **Global Configuration Mode** (Switch(config)#) door Switch# **Configure Terminal** te typen.

Er zijn ook nog sub-modes zoals Line Configuration Mode (om SSH, Telnet en AUX te configureren) en Interface Configuration Mode (om een switch port of een router network interface te configureren).

2.1.4: The command structure

The general syntax for a command is the command followed by any appropriate keywords and arguments.

**Keyword** - a specific parameter defined in the operating system (in the figure, ip protocols)

**Argument** - not predefined; a value or variable defined by the user (in the figure, 192.168.10.5)

2.2.1: Hostnames

Een eerste stap bij de configuratie van een netwerkapparaat is het geven van een unieke naam (anders krijgt een switch gewoon de naam ‘Switch’). Er zijn enkele regels die gevolgd moeten worden bij de naamgeving: Het moet starten met een letter, geen spaties, enkel letters, cijfers en dashes, max 64 karakters. Om een naam te geven moeten we in global configuration mode zijn. We schrijven dan: Switch(config)# **hostname** *naam*. Om een naam te verwijderen: no hostname.

2.2.2: Limit access to Device Configuration (Paswoorden)

Paswoorden worden gebruikt om de User EXEC-access, de Privileged EXEC-access en de Telnet access te beveiligen.

Om de privileged EXEC-access te beveiligen gebruiken we in global config mode: **enable secret** *paswoord*. Om de user EXEC-mode te beveiligen moeten we in console configuration mode zijn: **line console 0**, en dan schrijven we **password** *paswoord*. Daarna kunnen we weer in user EXEC-mode gaan met **login**.

To secure VTY lines used for SSH and Telnet, enter line VTY mode using the **line vty 0 15** global config command. Next, specify the VTY password using the **password** *password* command. Lastly, enable VTY access using the **login** command.

Alle ‘plain text’ passwoorden encrypteren: **service password-encryption**.

Banner Messages: To create a banner message of the day on a network device, use the **banner motd #** *the message of the day* **#** global config command.

2.2.3: Save configuration

Er bestaan 2 systeembestanden die de configuratie kunnen opslaan:

-Startup-config: Het bestand wordt in NVRAM opgeslaan die alle commando’s bevat dat het apparaat zal gebruiken als het opgestart wordt. NVRAM verliest geen gegevens als het apparaat uitstaat.

-Running-config: Het bestand opgeslaan in RAM die de configuratie die op het ogenblik gebruikt wordt toont. De running configuratie aanpassen veranderd de werking van de Cisco apparaat onmiddellijk.

Als het apparaat uitgaat verliezen we alles dat in de RAM staat, we kunnen het wel opslaan in de NVRAM met de commando: **copy running-config startup-config**. (afgekort: copy run start)

Als we de startup-config willen verwijderen gebruiken we de commando **erase startup-config**.

2.3 Address schemes

2.3.1: Ports and addresses

Verschillende apparaten hebben een IP-adres (Internet Protocol) nodig.

Het belangrijkste reden waarom IP-adressen gebruikt moeten worden is om het mogelijk te maken voor toestellen om andere apparaten te lokaliseren en ermee te communiceren op het internet.

Een soort van IP-adres is IPv4. Het bestaat uit 4 keer een cijfer tussen 0 en 255. IPv4 adressen worden toegekend aan apparaten die aan een netwerk zijn verbonden. Met een IPv4-adres hebben we ook een subnet mask nodig. We weten hiermee op welke subnet het apparaat zich bevindt.

SVI: A virtual interface, because there is no physical hardware on the device associated with it. An SVI is created in software. (Bv: Een VLAN interface).

For a switch to have an IP address, a **switch virtual interface** must be configured. This allows the switch to be managed remotely over the network.

2.3.2: Configure IP Addressing

Voordat een apparaat kan communiceren over een netwerk moet het een IP-adres en subnet mask krijgen. Dat kan automatisch gebeuren met DHCP (Dynamic Host Configuaration Protocol) of dat kan manueel.

Als we een switch willen beheren van ver, dan moet deze switch een IP-adres hebben. We kunnen dit doen vanaf de global config mode met de commando **interface vlan 1** (vlan = virtual lan)en daarna een IP adres te configureren met de commando **ip address** *ip-address subnet-mask.* Standaard de interface vlan shutdown, om de interface te activeren gebruiken we de commando **no shutdown**.

2.3.3: Verifying Connectivity

In the same way that you use commands and utilities like **ipconfig** to verify a PC host’s network configuration, you also use commands to verify the interfaces and address settings of intermediary devices like switches and routers. Command: **show ip interface brief**

End-to-end connectivity test: **ping** command.

# Hoofdstuk 3: Network Protocols and Communications

3.1.1: The rules of communication

Waarvoor dienen protocollen: De zender en krijger van data te indentificeren, een algemene taal, snelheid en timing van levering en confirmatie en bevestiging van vereisten.

Message encoding: The process of converting information into another acceptable form, for transmission. Decoding reverses this process in order to interpret the information.

Regels voor ‘message timings’:

-**Access Methods**: Bepaalt wanneer iemand kan communiceren. Als iedereen tegelijk communiceert wordt het onverstaanbaar.

# -**Flow Control**: De communicatie mag ook niet te snel gebeuren. ‘Source’ en ‘destination’ hosts gebruiken flow control om een correcte timing te bepalen voor een succesvolle communicatie.

# -**Response Timeout**: Hosts op netwerken hebben regels om te bepalen hoe lang ze zouden moeten wachten op een antwoord en wat ze moeten doen als ze geen antwoord krijgen.

# Berichten kunnen op 3 verschillende manieren worden geleverd op een netwerk, **unicast**, waar er enkel 1 bestemming is voor het bericht, **multicast**, waar er meerdere bestemmingen zijn en **broadcast** waar alle hosts op een netwerk het bericht krijgen.

# 3.2: Network protocols and standards

# 3.2.1: Protocols

# Belangrijke protocols voor de interactie tussen een webserver en een web client:

# **HTTP** - is an application protocol that governs the way a web server and a web client interact. HTTP defines the content and formatting of the requests and responses that are exchanged between the client and server.

**TCP** - is the transport protocol that manages the individual conversations. TCP divides the HTTP messages into smaller pieces, called segments. These segments are sent between the web server and client processes running at the destination host. TCP is also responsible for controlling the size and rate at which messages are exchanged between the server and the client.

**IP** - is responsible for taking the formatted segments from TCP, encapsulating them into packets, assigning them the appropriate addresses, and delivering them to the destination host.

**Ethernet** - is a network access protocol that describes two primary functions: communication over a data link and the physical transmission of data on the network media. Network access protocols are responsible for taking the packets from IP and formatting them to be transmitted over the media.

3.2.2: Protocol Suites

# Protocol suite: Een groep inter-gerelateerde protocols, nodig om een bepaalde taak uit te voeren.

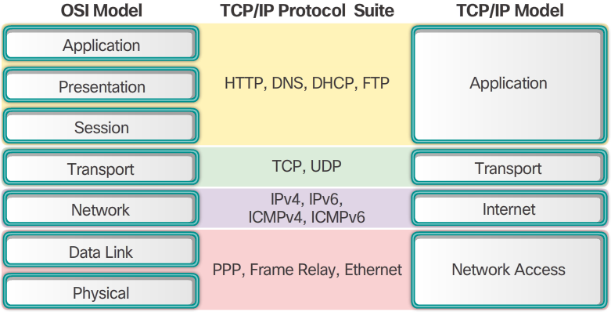
# TCP/IP is de meest gebruikte protocol suite. Transmission Control Protocol en Internet Protocol.

# 3.2.3: Standard Organisations

Internet Standards: Standards organizations are usually vendor-neutral, non-profit institutions established to develop and promote the concept of open standards. Various organizations have different responsibilities for promoting and creating standards for the TCP/IP protocol.

3.2.4: Reference Models

**De OSI Reference Model**:

7 lagen: (**A**ll **P**eople **S**eem **T**o **N**eed **D**ata **P**rocessing).

Andere richting: **P**lease **D**o **N**ot **T**hrow **A**way **S**ausage **P**izza

**Application**: specificeert, waarmee computers met elkaar kunnen communiceren over netwerken.

**Presentation**: Vertaalt en formatteert de data en levert deze aan de application layer voor verdere behandeling of weergave.

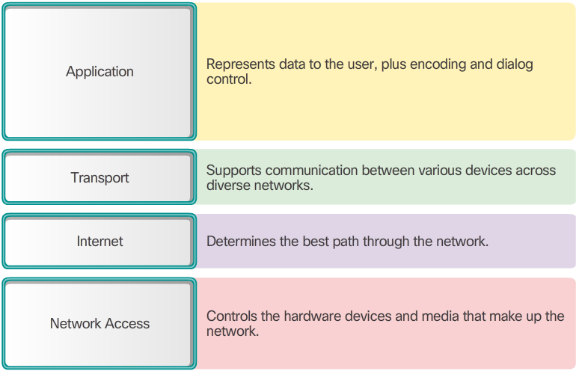
**Session**: regelt de totstandkoming, onderhoudt en beëindigt een sessie tussen twee communicerende hosts.

**Transport**: zorgt voor het probleemloze transport van data voor de applicaties. (TCP, UDP)

**Network**: verantwoordelijk voor het routeren van de pakketten door het netwerk**.** Encapsuleert IP-adres

**Data Link**: zorgt voor transport van de data over een verbinding (link).(Encapsuleerd MAC-Adres)

**Physical**: Deze bevat de elektrische en mechanische definities van transportmedium en het signaal. (Vertaalt ook binaire informatie in een signaal).



TCP/IP Protocol Model:

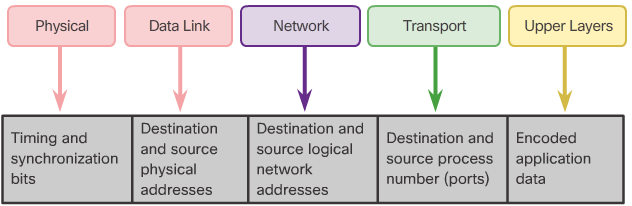
3.3.1: Data Encapsulation

Message Segmentation: Om het versturen van data te vergemakkelijken wordt data in kleinere pakketjes opgedeeld (= **segmentation**). Op deze manier kunnen meerdere mensen tegelijkertijd data sturen door overlapping (= **multiplexing**), en als er ergens een probleem is, dan moet enkel 1 pakket opnieuw worden verstuurd.

Encapsulation: Het process waarbij data door elke laag van het model gaat en bij elke laag protocol informatie wordt toegevoegd. De vorm van een stuk data na elke laag is een protocol data unit (PDU).

De-capsulation: De encapsulation process, maar omgekeerd bij de ontvangende host.

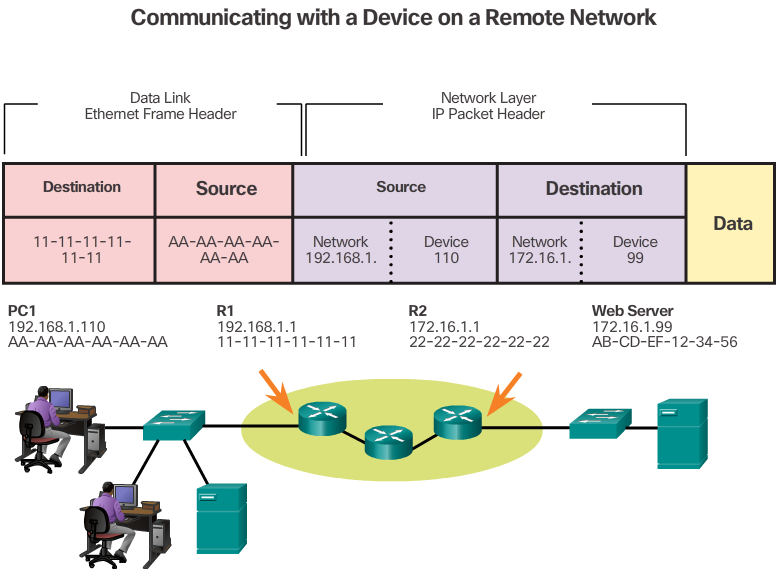
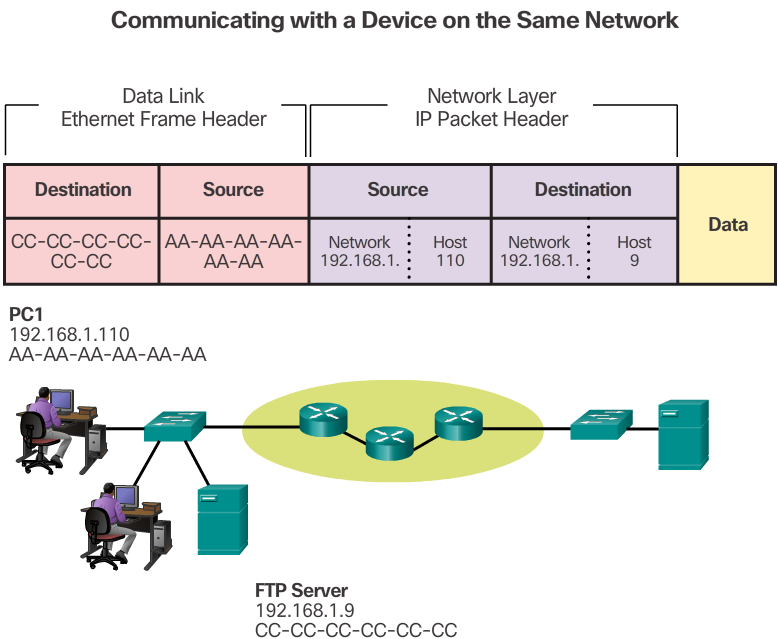
3.3.2: Data Access

De network en data link lagen zijn verantwoordelijk voor het leveren van data van de ‘source device’ tot de ‘destination device’. Het protocol van beide lagen hebben een source en destination adres, maar hebben verschillende doeleinden:

**Network layer source and destination addresses:** Responsible for delivering the IP packet from the original source to the final destination, either on the same network or to a remote network.

**Data link layer source and destination addresses** – Responsible for delivering the data link frame from one network interface card (NIC) to another NIC on the same network. (Met MAC-adressen).

IP-adres = network layer. Een IP packet heeft 2 IP-adressen, het IP-adres van de source en destination host.



|  |  |
| --- | --- |
| Role of the Network Layer Addresses:  The network layer addresses, or IP addresses, indicate the original source and final destination. An IP address contains two parts:  -Network portion  -Host portion | Role of the Network Layer Addresses:  When the sender of the packet is on a different network from the receiver, the source and destination IP addresses will represent hosts on different networks. This will be indicated by the network portion of the IP address of the destination host. |
| Role of the Data Link Layer Addresses:  When the sender and receiver of the IP packet are on the same network, the data link frame is sent directly to the receiving device. On an Ethernet network, the data link addresses are known as Ethernet (MAC) addresses. MAC addresses are physically embedded on the Ethernet NIC.  -Source MAC address - This is the data link address, or the Ethernet MAC address, of the device that sends the data link frame with the encapsulated IP packet.  -Destination MAC address - When the receiving device is on the same network as the sending device, this is the data link address of the receiving device. | Role of the Data Link Layer Addresses:  When the sender and receiver of the IP packet are on different networks, the Ethernet data link frame cannot be sent directly to the destination host because the host is not directly reachable in the network of the sender. The Ethernet frame must be sent to another device known as the router or default gateway.  -Source MAC address - The Ethernet MAC address of the sending device, PC1.  -Destination MAC address - When the receiving device, the destination IP address, is on a different network from the sending device, the sending device uses the Ethernet MAC address of the default gateway or router. |

# Hoofdstuk 4: Network Access

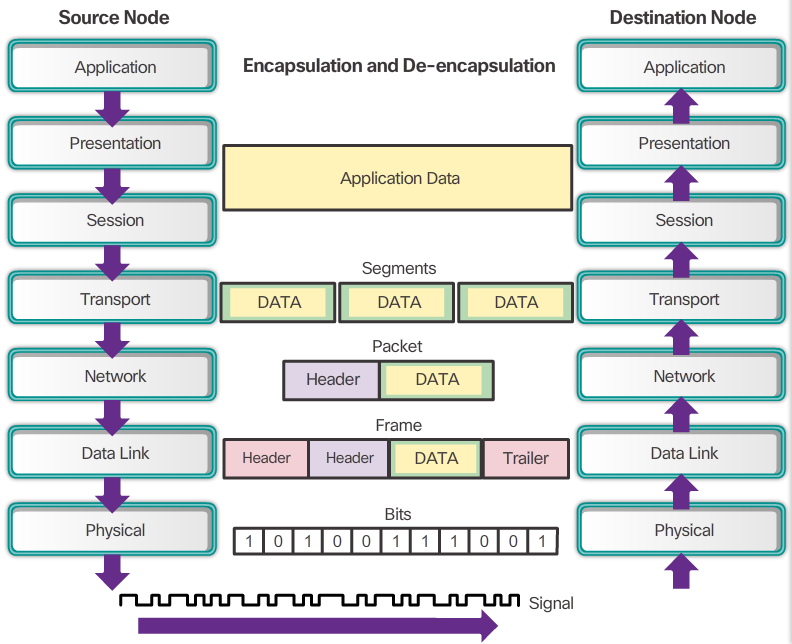
! Media is the physical copper cable, optical fiber, or atmosphere through which the signals travel.

4.1.1 : Physical Layer Connection

Physical connection: Voordat we kunnen communiceren door middel van onze end-devices, moeten we ze verbinden met het lokale netwerk, dit kan zowel gekabeld als draadloos gebeuren. Om het mogelijk te maken draadloze toestellen te gebruiken, hebben we een Wireless Access Point (AP) nodig.

AP en switchen zijn vaak 2 verschillende apparaten in een netwerk, maar er bestaan ook routers die AP en switch samen integreren. **ISR** of Integrated Service Router.

4.1.2: Purpose of the physical layer

De physical layer van het OSI-model biedt het middel aan om bits (waaruit ons data bestaat) te transporteren doorheen een netwerk.

Proces dat data ondergaat van source tot destination node:

1. De data wordt gesegmenteerd door de transport laag, geplaatst in packets door de netwerk laag en verder geëncapsuleerd in frames door de data link laag.

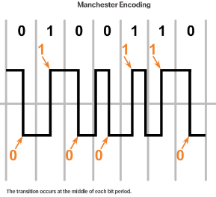
2. De physical layer encodeerd de frames en maakt de elektrische, optische, of radiogolf signalen die bits voorstellen in elke frame.

3. Deze signalen worden een voor een op de media gestuurd.

4. De physical layer van de destination node pakt deze signalen op van de media en zet ze weer om tot hun voorstelling in bits. Daarna geeft het de bits door aan de data link laag als een complete frame.

Er bestaan 3 vormen van netwerk media: **Koperen kabel**: de signalen zijn patronen van elektrische pulsen; **Fiber-optic kabel**: de signalen zijn patronen van licht; **Wireless**: de signalen zijn patronen van microgolf-uitzendingen.

De standaarden van de physical layer zijn geïmplementeerd in de hardware. Voor alle andere lagen is dat in de software.

4.1.3: Physical Layer Characteristics

Manchester encoding.

The physical layer standards address three functional areas:

**Physical components:** De elektronische hardware apparaten die de signalen overdragen en zenden die bits voorstelen.

**Encoding:** De methode waarop de ‘stream’ van bits wordt geconverteerd in een code. (vb: Manchester Encoding).

**Signaling:** Hoe de physical layer een elektrische, optische of wireless signaal genereerd die 1 en 0 voorstellen (bits).

Bandwith: Het maximaal aantal data (bits) dat per seconde van de ene plaats naar de andere kan stromen, uitgedrukt in kb/s, Mb/s, Gb/s.

Throughput: De hoeveelheid data dat gedurende een bepaalde periode van de ene plaats naar de andere echt stroomt, troughput is vaak lager dan de bandwith door verschillende factoren. (latency, type of traffic, amount of traffic).

4.2.1: Copper Cabling

Koperen kabels worden veel gebruikt, maar hebben nadelen, hoe langer de afstand dat het signaal moet afleggen, hoe slechter dat signaal wordt. Het signaal kan ook slechter worden door storing van 2 verschillende bronnen:

**Electromagnetic interference (EMI) of radio frequency interference (RFI):** Radiogolven en elektromagnetische golven kunnen de signalen die door koperen kabels gaan verpesten.

Om deze effecten tegen te gaan worden sommige koperen kabels ingepakt in een beschermende metalische laag.

**Crosstalk:** Als koperen kabels te dicht bij elkaar liggen, kan je crosstalk hebben, tijdens een telefoongesprek kan je dan bijvoorbeeld een andere conversatie vanop een andere lijn horen.

Om crosstalk tegen te gaan, hebben sommige koperen kabels kabels die getwist worden.

3 soorten koperen kabels: **Unshielded Twisted-Pair (UTP), Shielded Twisted-Pair (STP) en Coaxial**.

De meest voorkomende zijn de **UTP-kabels** omdat ze goedkoop zijn. Het feit dat ze getwist zijn beschermd ze tegen interferentie (storing) en dus tegen crosstalk. Ze bestaan uit 4 paren van kleur gecodeerde kabels die samen getwist zijn.

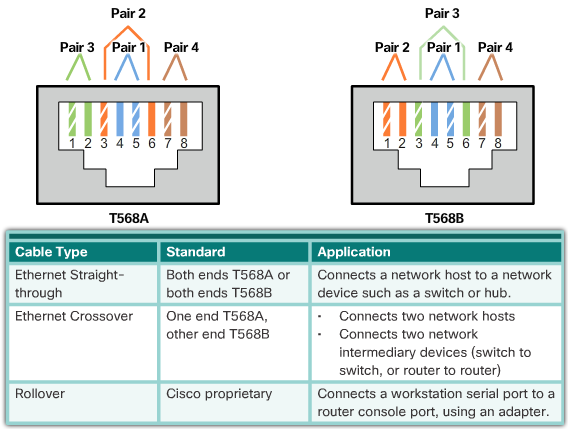
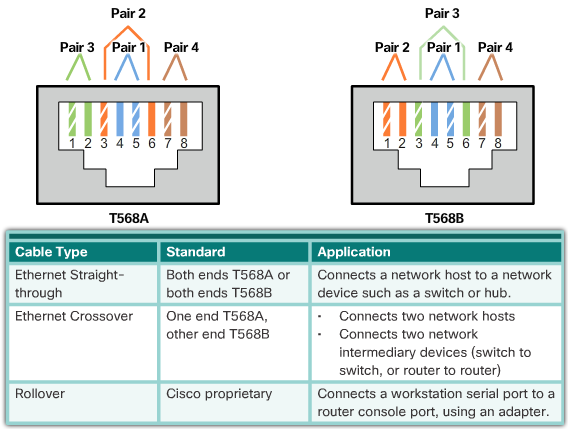
**STP**-kabels worden minder gebruikt omdat ze duurder zijn, maar ze zijn wel beter beschermd tegen ruis. Ze hebben ook metalische shielding tegen EMI en RFI.

**Coaxial**-kabels bestaan uit een koperen kern, met een isolator van plastic en daarrond een koperen mantel. Het geheel wordt net zoals andere kabels omhuld door nog een laag plastic.

4.2.2: UTP Cabling

UTP kabels hebben vaak een RJ45 connector.

Er bestaan 3 soorten UTP-kabels. Straight-through, Ethernet Crossover en Rollover.

4.2.3: Fiber-Optic Cabling

Een Fiber-optic kabel kan data sturen over langere afstanden en met een hogere bandbreedte dan koperen kabels. Het is ook immuun voor EMI en RFI. Het bestaat uit 2 verschillende soorten glas, één voor de kern en een andere rond de kern dat het licht als een spiegel terug naar de kern weerkaatst.

Er bestaan 2 soorten optic-fiber kabels:

**Single-Mode:** Licht komt uit lasers, smalle kern, minder verspreiding, licht gaat door de smalle kern rechtstreeks naar de eindbestemming. Wordt gebruikt voor langere afstanden.

**Multimode:** Licht komt uit LEDs, bredere kern, meer verspreiding 🡪 meer signaalverlies, licht wordt op spiegeltjes weerkaatst naar eindbestemming. Lange afstand, maar korter dan Single-Mode.

Licht kan enkel in 1 richting gaan, daardoor bestaan fiber-optic kabels uit 2 kabels, 1 voor elk richting.

4.2.4: Wireless Media

Wireless media draagt elektromagnetische signalen over die de binaire getallen van datacommunicatie voorstellen door het gebruik van radio of microgolf frequenties.

De nadelen van draadloze media zijn:

**Coverage area:** Muren, deuren, etc. maken de signalen van draadloze media minder sterk.

**Interference:** Draadloze media zijn vatbaar voor interferentie.

**Security:** Apparaten kunnen zonder autorisatie het netwerk binnenkomen.

**Shared Medium:** Hoe meer mensen verbonden zijn aan 1 WLAN, hoe trager de connectie omdat de bandwidth voor elk gebruiker kleiner wordt.

Soorten wireless media: Wi-Fi (IEEE 802.11), Bluetooth (IEEE 802.15), WiMax (IEEE 802.16).

4.3.1: Purpose of the Data Link Layer

Data Link is de tweede laag van het OSI-model. Het heeft verschillende taken:

-Allowing the upper layers to access the media

-Accepting Layer 3 packets and packaging them into frames

-Preparing network data for the physical network

-Controlling how data is placed and received on the media

-Exchanging frames between nodes over a physical network media, such as UTP or fiber-optic

-Receiving and directing packets to an upper layer protocol

-Performing error detection

De data link layer verdelen we onder 2 verschillende sub-lagen: De LLC en de MAC.

**Logical Link Control (LLC):** Communiceert met het netwerk laag. Het plaatst informatie in de frame dat herkent welke protocol gebruikt wordt voor dat frame.

**Media Access Control (MAC)**

4.4.1: Topologies

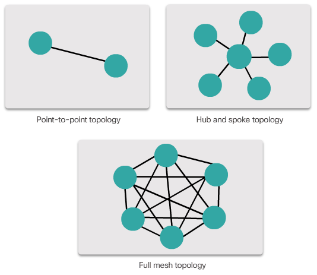
Media access control is te vergelijken met de wetten op straat, wanneer mag een auto op straat rijden? Het moet oppassen voor het andere verkeer.

Met **topologie** bedoelt men hoe apparaten binnen een netwerk gekoppeld worden. LAN en WAN topologieën kunnen op 2 manieren gezien worden. De physical topology en de logical topology.

**Physical Topology:** De structuur van de kabels binnen een netwerk. Vb. mesh, point-to-point, hub.

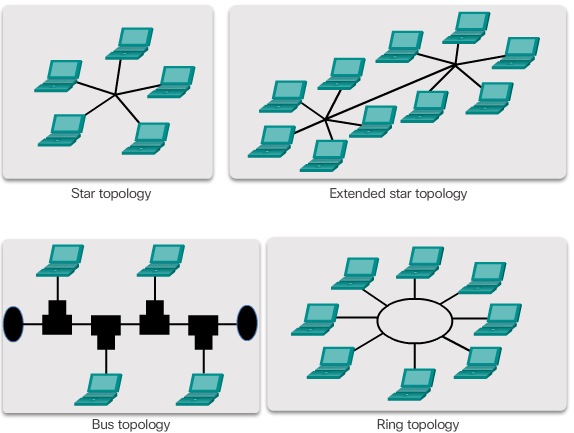
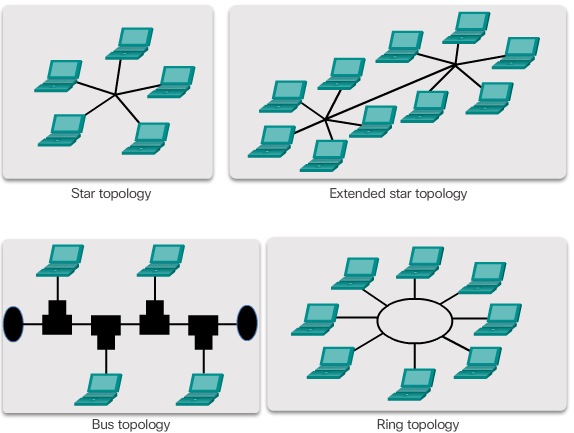
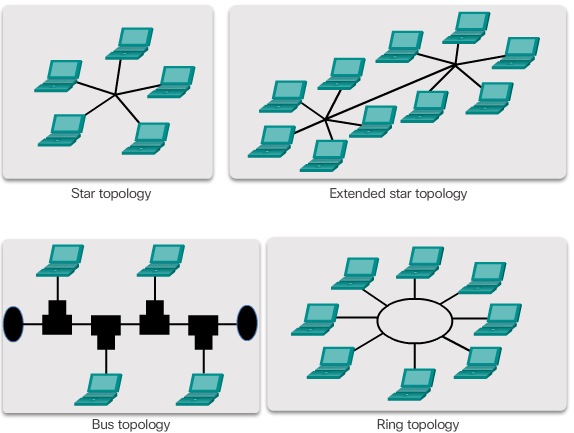
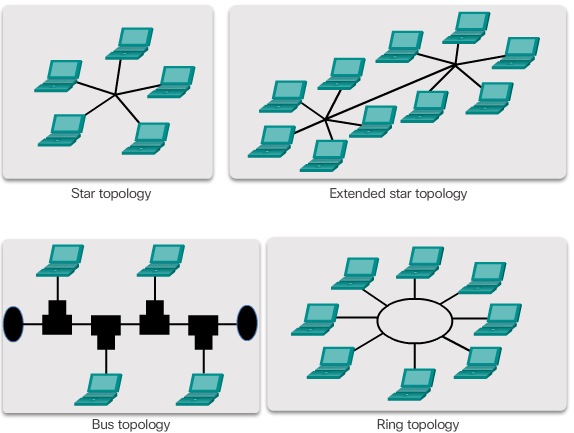
**Logical Topology:** Bijvoorbeeld de netwerkprotocollen, die instaan voor de manieren waarop computers met elkaar communiceren. Het bestaat uit virtuele connecties tussen de apparaten op het netwerk (computers, netwerkapparaten…).

4.4.2: WAN Topologies

**Point-to-Point:** Simpelste topologie, bestaat uit een permanente link tussen twee eindpunten.

**Hub and Spoke:** Sterrentopologie voor WANs, alle apparaten zijn verbonden aan één apparaat in het midden.

**Mesh:** Elk eindsysteem is verbonden aan elk ander apparaat (via point-to-point). Het is de beste topologie maar het kost enorm veel.

4.4.3: LAN Topologies

**Star:** Alle toestellen worden aangesloten op een centraal punt (vaak een switch).

**Extended Star:** Meerdere sternetwerken samen verbonden.

**Bus:** Alle computers zijn aangesloten op dezelfde kabel, de kabel heeft een begin- en eindpunt. De computers communiceren met elkaar via één enkele kabel van computer tot computer.

**Ring:** Elk computer is verbonden aan een ‘buur’ via een kabel, zodat het uiteindelijk een ring vormt.

Duplex communication: Verwijst naar de richting waarin data wordt uitgezonden tussen twee apparaten. We praten over **half-duplex communicatie** (enkel in 1 richting tegelijk) en **full-duplex communicatie** (zenden en ontvangen van data gebeurt tegelijkertijd).

Alle apparaten die in half-duplex werken moeten concurreren om data te mogen sturen. Daarom bestaat **controlled access**, waar elk apparaat een tijd krijgt om iets te sturen. Maar dit is niet efficiënt.

De Carrier Sense Multiple Access/Collision Detection (**CSMA/CD**) process wordt gebruikt in half-duplex Ethernet LANs. Dit is een toegangsprotocol dat bepaalt wanneer netwerkapparaten pakketten mogen beginnen verzenden. CSMA/CD bevindt zich op de datalink laag van het OSI-model. 🡪 Detecteert botsingen.

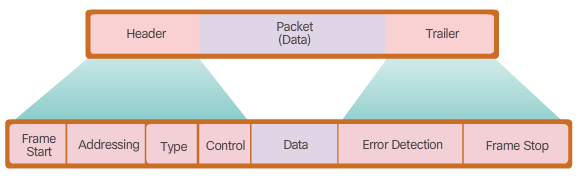
Een ander vorm van CSMA is **CSMA/CA** dat gebruikt wordt bij WLANs. Het kan collisions wel niet detecteren maar probeert ze te vermijden door te wachten voor het uitzenden.

4.4.4: Data Link Frame

De Data Link Laag bereid een packet voor voor het transport door de locale media door het te enapsuleren om een frame te maken. Een frame bestaat uit een **Header**, **Data** en **Trailer**.

Frame Fields:

**-Frame start and stop indicator flags**: Used to identify the beginning and end limits of the frame.

**-Addressing**: Indicates the source and destination nodes on the media.

**-Type**: Identifies the Layer 3 protocol in the data field.

**-Control**: Identifies special flow control services such as quality of service. QoS is used to give forwarding priority to certain types of messages. (Streaming / webpagina’s)

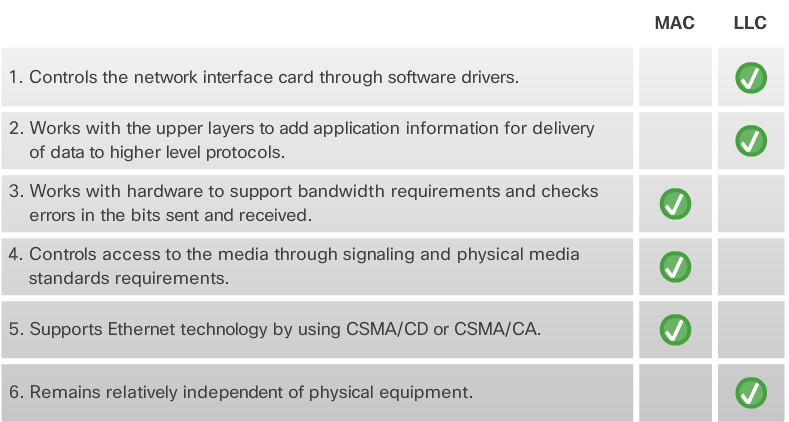
**-Data**: Contains the frame payload (i.e. packet header, segment header, and the data).

**-Error Detection**: These frame fields are used for error detection and are included after the data to form the trailer.

# Hoofdstuk 5: Ethernet

5.1.1: Ethernet Frame

Ethernet is de meest gebruikte LAN-technologie. Het werkt in de physical en data-link layer van het OSI-model. Ethernet = IEEE 802.2 (LLC) en 802.3 (MAC) standaarden. Ethernet steunt dus op de 2 sublayers van de data link layer, nl. LLC en MAC.

LLC Sublayer: Zorgt voor de communicatie tussen de bovenste lagen en de onderste lagen. Het neemt de netwerkprotocol data (vaak een IPv4 packet) en voegt er ‘control informatie’ aan toe om te helpen bij het leveren van het packet naar het eindapparaat.

MAC Sublayer: Zorgt voor data encapsulation en Media Adressing Control:

Data Encapsulation: The data encapsulation process includes frame assembly before transmission, and frame disassembly upon reception of a frame. In forming the frame, the MAC layer adds a header and trailer to the network layer PDU (Protocol Data Unit).

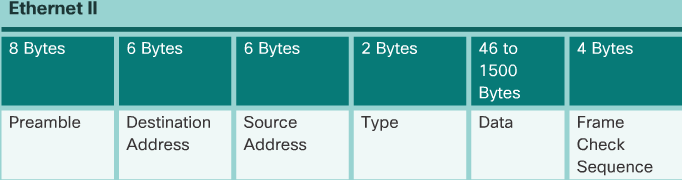
Functies van Data Encapsulation:

**Frame Delimiting**: Determineren welke groep van bits deel uitmaken uit een frame.

**Addressing**: De encapuslatie proces bevat de Layer 3 PDU en voorziet ook een data-link layer adressering.

**Error Detection**: Elke frame heeft een ‘trailer’, die gebruikt wordt om fouten in de transmissie te vinden.

Media Access Control: Communiceert met de physical layer. Het is verantwoordelijk voor het plaatsen van de frames op de media (fysieke kabel) en ook het weghalen van de frames van op de media. Het controleert dus de toegang tot de media.

De minimum ethernet frame = 64bytes, maximum ethernet frame = 1518 bytes (zonder preamble).

- Kleiner dan 64 bytes = Collision fragment, groter dan 1518 bytes = jumbo. Deze worden meteen gedropt door de ontvangende apparaat.

Functie van elk ethernet II frame fields:

**Preamble + start of frame delimiter:** Zorgt voor synchronisatie tussen de zender en de ontvangende apparaat. Vertelt de ontvanger dat hij zich klaar moet maken om een frame te ontvangen.

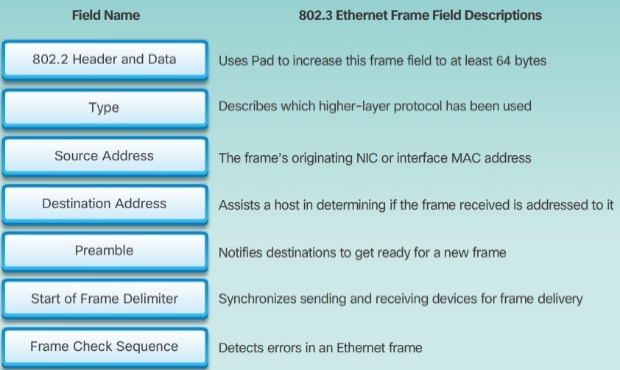
**Destination Address:** Identificeert de ontvanger. Het adres in de frame wordt vergeleken met het MAC-adres van de ontvanger. Als ze overeenstemmen, dan accepteert de ontvangende apparaat de frame.

**Source Address:** Identificeert de NIC van de zender. Is altijd unicast. (destination address kan ook multicast of broadcast zijn).

**Type:** Identificeert de upper layer protocol dat in de frame geëncapsuleerd is.

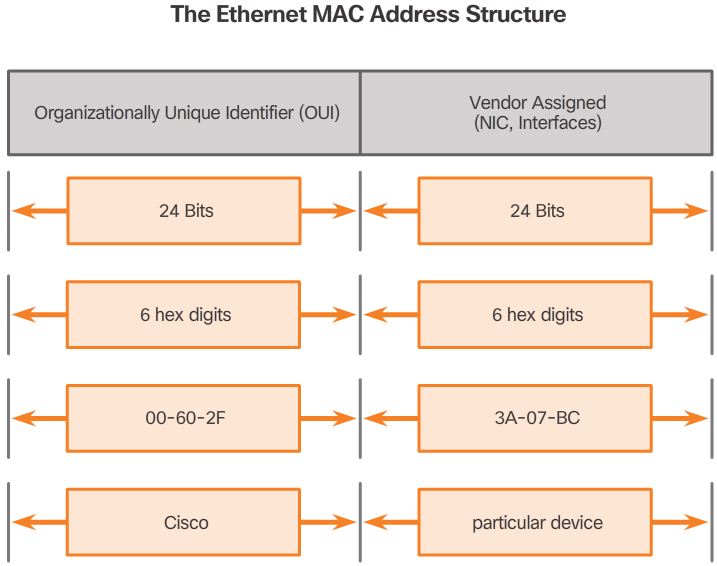
**Data:** Bevat de geëncapsuleerde data uit een hogere laag, het is vaak een IPv4 packet. Als een frame kleiner dan 64 bytes geëncapsuleerd wordt, dan worden er extra bits (pads) gebruikt om de frame groter te maken.

**Frame Check Sequence:** Wordt gebruikt om fouten in de frame te vinden. Als er een fout is wordt de frame gedropt.



5.1.2: Ethernet MAC addresses

Een ethernet MAC-adres bestaat uit een 48-bit binaire waarde, uitgedrukt als 12 hexadecimale getallen (4 bits = 1 hexadecimaal getal.). 8 bits = 1 byte, in hexadecimaal 🡪 van 00 tot FF. In tekst is een hexadecimaal getal vaak achter 0x. Vb. 0x73 of 0x0A. Hexadecimale notatie wordt ook gebruikt voor IPv6 adressen.

IEEE requires a vendor to follow two simple rules, as shown in the figure:

-All MAC addresses assigned to a NIC or other Ethernet device must use that vendor's assigned OUI (Organizationally Unique Identifier) as the first 3 bytes.

-All MAC addresses with the same OUI must be assigned a unique value in the last 3 bytes.

Note: It is possible for duplicate MAC addresses to exist due to mistakes during manufacturing or in some virtual machine implementation methods. In either case, it will be necessary to modify the MAC address with a new NIC or in software.

De MAC-adres is in de ROM ‘ingebrand’, wanneer een computer boot, wordt de MAC-adres in de RAM gestopt. Wanneer een bericht wordt verzonden op een netwerk, staat de source en destination MAC-adres in de header-informatie van het packet. Wanneer een NIC een packet krijgt, vergelijkt hij de destination adres, met zijn eigen adres, als ze niet overeenkomen, wordt dat packet gedumpt.

Unicast MAC-adres: Een uniek MAC-adres, voor wanneer we een frame enkel naar één host sturen.

Het process om het destination MAC-adres te bepalen noemt het **Address Resolution Protocol** (**ARP**).

De destination MAC-adres kan unicast, multicast of broadcast zijn, maar de source is altijd unicast.

Broadcast MAC-adres: FF-FF-FF-FF-FF-FF (48 enen in binair).

Multicast MAC-adres: Het multicast MAC-adres is een special waarde dat begint met 01-00-5E in hexadecimaal.

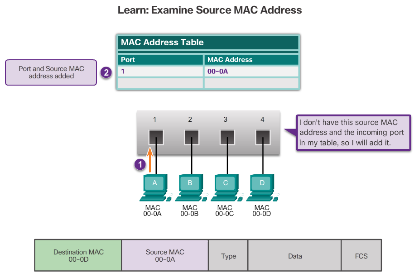
5.2: LAN switches

5.2.1: The MAC Address Table

Een switch is een toestel van de 2de laag (data link), het gebruikt de MAC-adres om forwarding beslissingen te nemen.

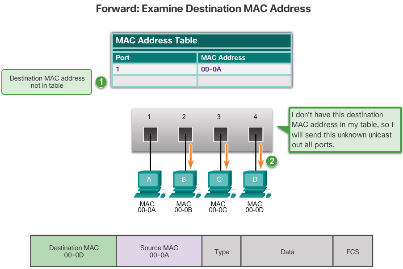
Hoe maakt een switch een MAC-adres table?

**Learn – Examining the Source MAC Address**

Elke frame de een switch binnenkomt wordt gecheckt voor nieuwe informatie. Dat gebeurt door de source MAC-adres van de frame te onderzoeken, en de poort waardoor de frame de switch binnenkwam.

* Als de source MAC-adres nog niet bestaat in de table, wordt het toegevoegd, samen met de poort nummer. *In Figure 1, PC-A is sending an Ethernet frame to PC-D. The switch adds the MAC address for PC-A to the table.*
* Als de source MAC-adres al bestaat, dan wordt de refresh timer voor dat MAC-adres geüpdatet. Standaard is dat in de meeste switchen 5 minuten voor elke MAC-adres.

**Forward – Examining the Destination MAC Address**

Als de destination MAC-adres unicast is, zal de switch zoeken naar een match in de MAC address table.

* Is de destination MAC-adres in de tabel, dan forward de switch de frame naar de juiste poort.
* Is het niet in de tabel, dan forward de switch de frame naar alle poorten, behalve de source poort (unknown unicast). *The switch does not have the destination MAC address in its table for PC-D, so it sends the frame out all ports except port 1.*

Wanneer een switch een frame krijgt van een apparaat, kan het de MAC-adres tabel aanvullen door de source MAC-adres van elke frame te examineren. Eens de de MAC address tabel een destination adres bevat, kan het een frame meteen naar de correcte poort sturen.

Zie video op netacad 🡪 5.2.1.4

5.2.2: Switch Forwarding Methods

Een switch gebruikt één forward-methode (schakelaar methode), **store-and-forward** **switching** of **cut-through switching**.

**Store-and-forward**: When the switch receives the frame, it stores the data in buffers until the complete frame has been received. During the storage process, the switch analyzes the frame for information about its destination. In this process, the switch also performs an error check using the Cyclic Redundancy Check (CRC) trailer portion of the Ethernet frame. When an error is detected in a frame, the switch discards the frame. Discarding frames with errors reduces the amount of bandwidth consumed by corrupt data. Store-and-forward switching is required for Quality of Service (QoS) analysis on converged networks where frame classification for traffic prioritization is necessary.

**Cut-through**: Forwards the frame begore it is entirely received. At a minimum, the destination address of the frame must be read before the frame can be forwarded.

Er zijn twee soorten cut-through switching: **Fast-forward switching** (snelste, maar kan packets doorsturen met fouten)en **Fragment-free switching** (een soort mengeling tussen fast-forward en store-and-forward switching, het zoekt naar fouten op de eerste 64 bytes, waar de meeste fouten en collisions gebeuren).

Memory Buffering van switchen: Een ethernet switch kan een buffering techniek gebruiken om frames op te slaan voor dat ze geforward worden. Buffering kan ook gebruikt worden wanneer de destinatie poort verstopt is door te veel verkeer. De switch slaat dan de frame op tot het gestuurd kan worden.

**Port-base memory**: Alle frames worden opgeslagen in rijen dat gelinkt zijn tot specifieke ‘incoming en outgoing’ poorten.

**Shared memory**: Alle frames worden in een gemeenschappelijk memory buffer opgeslagen, die gedeeld wordt tussen alle poorten op de switch.

5.2.3: Switch Port Settings

Er zijn 2 belangrijke settings voor een switch: de duplex en de bandwidth. Deze settings moeten dezelfde zijn tussen de switch poort en de geconnecteerde toestel.

Autonegotiation bestaat vandaag op de meeste switchen en NIC’s, hiermee kunnen twee apparaten automatisch informatie met elkaar delen over hun duplex en snelheid. De apparaten kiezen dan voor de snelst mogelijke bandbreedte en voor full-duplex als dit op beide apparaten mogelijk is.

5.3.1: MAC and IP

Op een LAN zijn er 2 belangrijke adressen die aan toestellen worden toegekend.

-Physical address (MAC-adres) –Om de data link frame met de geëncapsuleerde IP-packet te leveren van één NIC tot een andere op dezelfde netwerk.

-Logical address (IP-adres) – Gebruikt om een packet te sturen van source tot bestemming. De bestemming kan zich ook op een ander netwerk bevinden.

When the destination IP address is on a remote network, the destination MAC address will be the address of the host’s default gateway, the router’s NIC.

When the router receives the Ethernet frame, it de-encapsulates the Layer 2 information. Using the destination IP address, it determines the next-hop device, and then encapsulates the IP packet in a new data link frame for the outgoing interface. Along each link in a path, an IP packet is encapsulated in a frame specific to the particular data link technology associated with that link, such as Ethernet. If the next-hop device is the final destination, the destination MAC address will be that of the device’s Ethernet NIC.

Om de destination MAC-adres te leren, gebruikt een toestel **ARP**. ARP heeft twee functies:

- IPv4 adressen omzetten tot MAC-adressen.

- Een tabel maken met de IP-adressen in de lokale IP-subnet en de bijbehorende MAC-adressen.

Er wordt een tabel gemaakt met alle IP-adressen en hun MAC-adres, een ARP-cache. Als een apparaat de destination MAC-adres niet heeft, stuurt het met de destination IP-adres een ARP request (broadcast).

Zie videos: 5.3.2.3.

Wanneer een MAC-adres niet meer gebruikt werd voor een bepaalde tijd (bv. 2 minuten) dan wordt de entry ervan in de ARP-cache verwijderd. Om de ARP-tabel te zien op een router van Cisco gebruiken we de commando **show ip arp**. Op Windows gebruiken we de commando **arp -a**.

# Hoofdstuk 6: Network Layer

6.1.1: Network layer and communication

Het netwerk laag (OSI laag 3) maakt het mogelijk om data te wisselen tussen end-devices. Het maakt gebruik van 4 processen:

**Addressing end devices:** End devices moeten een uniek IP-adres hebben om geïdentificeerd te worden.

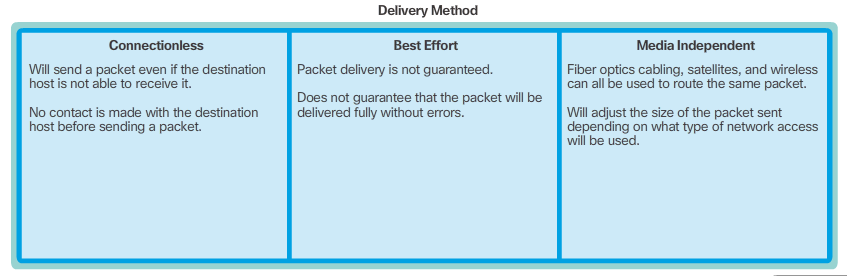
**Encapsulation:** De network laag encapsuleert de PDU van het transport laag tot een packet, hierbij wordt de source en destination IP-adres toegevoegd (IP-header information).

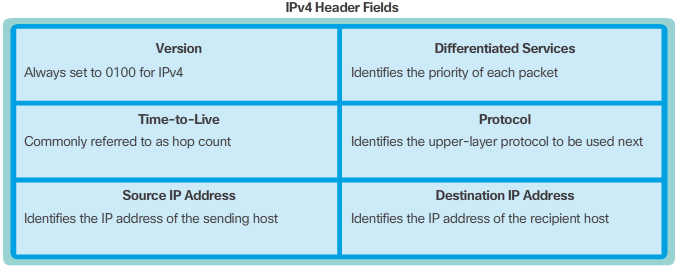
**Routing:** Het netwerk laag heeft ook als dienst de packets te leiden naar een ander netwerk. De packet gaat door een router, en een router moet de beste weg kiezen op een netwerk.

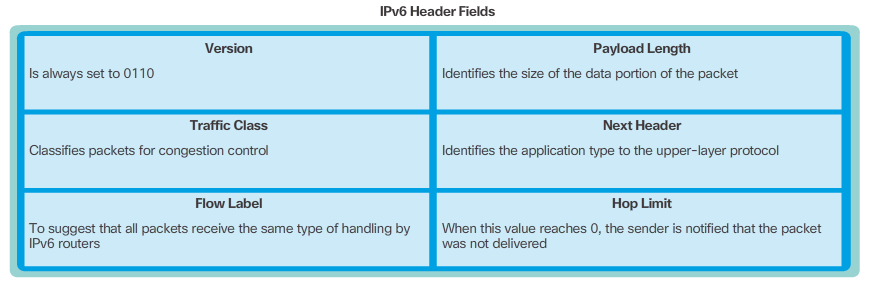
**De-Encapsulation:** Wanneer een packet op het netwerk laag van de destination host aankomt moet de host naar de IP-header information kijken. Als deze correct is, dan wordt het ge-decapsuleerd.

MTU: Maximum Transmission Unit.

Network layer protocollen: IPv4 en IPv6 (IP kan niet verzekeren dat packets echt aankomen tot hun bestemming).



Enkele belangrijke velden in de IPv4 packet header: **Version** (0100, versie 4); **Differentiated Services** (**DS,** geeft de prioriteit aan van een packet); **Time-to-Live** (TTL, een limiet op de levensduur van een packet. Een packet krijgt een beginwaarde, en elke keer dat het door een router gaat, doen we de waarde -1. Als de waarde nul wordt, krijgen we een Time Exceeded bericht); **Protocol** (geeft aan welk protocol gebruikt wordt om de data door te geven naar een hogeren laag, vb: TCP(6), UDP(17) of ICMP(1, Internet Control Message Protocol); **Source en Destination IP address**.

Velden uit de IPv6 packet header: **Version, Traffic Class** (=DS van IPv4), **Flow Label, Payload Length** (de lengte van de packet)**; Next Header** (= protocol veld van IPv4); **Hop Limit** (= TTL); **Source en Destination IP address.**

6.2.1: How a Host Routes

Een andere rol van het netwerk laag is om packets te leiden tussen de hosts. Een host kan een packet versturen naar: Zichzelf (speciale IPv4 Adres 127.0.0.1), een lokale host of een host op een ander netwerk.

Whether a packet is destined for a local host or a remote host is determined by the IPv4 address and subnet mask combination of the source (or sending) device compared to the IPv4 address and subnet mask of the destination device.

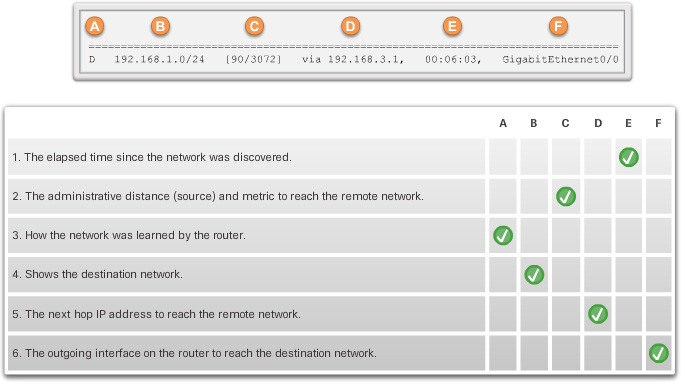
De Default gateway: Wordt gebruikt om verkeer naar andere netwerken te routen.

Host Routing Table: **netstat -r** of **rout print** om op windows te tonen. Op een Cisco router: **show ip route**

Kiezen waar een packet wordt verstuurd: Wanneer een host een bericht stuurt, gebruikt het de routing table om te bepalen waar het de packet naartoe moet sturen. Als de bestemming zich op een ander netwerk bevindt, wordt de packet verstuurd naar de default gateway. Vaak is dit de router, de router kijkt dan in zijn eigen routing table om te weten waar het gestuurd moet worden.  
Routing table van een router lezen: Zie video 6.2.2.7

Voor een directly-connected network: Eerste letter, **C** (directly-connected network); **L** (local interface); daarnaast staat de destination network en hoe we die kennen als laatste staat de exit interface (bv GigabitEthernet0/0).

Voor een remote network: **Letter** – destination network - … - via **volgende router** (next hop address) – laatste tijd dat de route werd geüpdatet – outgoing interface.



6.3.1: Anatomy of a Router

**CPU**; **RAM**: IOS, Configuration file, ARP table, Routing table, Packet buffer; **NVRAM**: Startup Configuration File; **Flash**: IOS Files; **ROM**: Power-on self-test, Backup van IOS.

De IOS en startup-config worden na de boot geladen in de RAM.

LAN and WAN intefaces: Console: Fysieke toegang tot de router, SSH: Secure Shell, veilige toegang tot router van ver, telnet: onveilige toegang tot router van ver (geen encryptie voor passwoorden etc.)

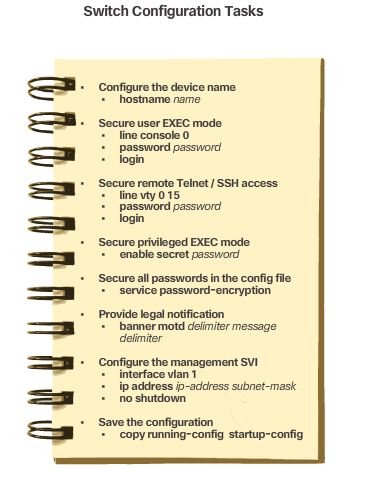
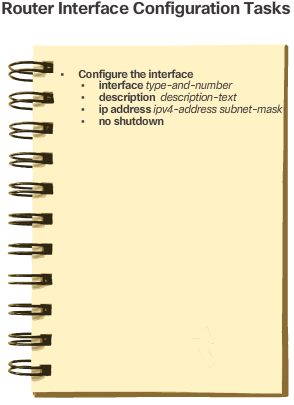
6.3.2: Router Boot-up

Bootup proces video: 6.3.2.3

POST – Load Bootstrap – Load IOS – Load Configuration File from FLASH (NVRAM) OR Go into Setup Mode.

Commando om router te rebooten: **reload**, om versie van IOS te tonen: **show version**. (Privileged exec)

6.4.1: Configure a Cisco Router: Configure Initial Settings + 6.4.2: Configure Interfaces

To verify the interface configuration:

**Show ip interface brief**

Verify connectivity: **Ping**

**show ip route** - Displays the contents of the IPv4 routing table stored in RAM.

**show interfaces** - Displays statistics for all interfaces on the device.

**show ip interface** - Displays the IPv4 statistics for all interfaces on a router.

6.4.3: Configure the Default Gateway

Global configuration command: **ip-adres default-gateway** *default gateway*.

# Hoofdstuk 7: IP Addressing

7.1.1 : Binary and Decimal Conversion

Een IP-adres bestaat uit 32 bits, verdeeld in 4 octetten, elk octet bestaat uit 8 bits (1 byte = 255).

7.1.2: IPv4 Address Structure

De eerste octetten zijn van de **Network Portion**, deze moeten binnen dezelfde netwerk voor alle apparaten dezelfde zijn. De laatste octet(ten) zijn van de **Host Portion**, deze moet voor elk apparaat verschillend zijn om de verschillende hosts op een netwerk te kunnen identificeren. De subnet mask zorgt ervoor dat de hosts weten welk deel van de Network Portion is en welk deel van de Host Portion is.

Wanneer we een adres configureren op een host, moeten we 3 IPv4 adressen ingeven: de IP-adres, de subnet mask en de default gateway (IPv4 adres van de router).

De 1-en van de subnet mask geven aan wat de Network Portion is, de 0-en wat de Host Portion is.

Het proces van het identificeren van de Network-en Host Portion noemen we **ANDing**. (1 AND 1 = 1,

0 AND 1 = 0, 0 AND 0 = 0) Door het IP-adres met de subnet mask te ANDen, vinden we de **Netwerk Adres**.

De lengte van het prefix: **/n** na de 32-bit adres. Het stelt het aantal 1-en in de subnet mask voor.

De laatste host portion is 254 (11111110), omdat 255 gereserveerd wordt voor broadcast.

Zie video 7.1.2.7

7.1.3: IPv4 Unicast, Broadcast and Multicast

Apparaten kunnen een statisch of dynamisch (DHCP of Dynamic Host Configuration Protocol) IP-adres krijgen. Sommige apparaten zoals printers, servers en netwerkapparaten hebben een statisch adres nodig.

**Unicast**: Host-to-host communicatie, het gebruikt het adres van de destination device als destination address. Range of IPv4 unicast host addresses: 0.0.0.0 – 223.255.255.255

**Broadcast**: Gebruikt om packets naar alle host van het netwerk te sturen. De destination IP-adres heeft dan in de Host Portion enkel 1-en. Alle hosts krijgen en kijken naar het packet. Er bestaan twee soorten broadcasts, **limited** en **directed broadcasts**. Een **directed** broadcast wordt gestuurd naar alle hosts op een bepaald netwerk (vb. 182.16.4.255). Een **limited** broadcast wordt gestuurd naar 255.255.255.255. Standaard word teen broadcast niet ge-forward door routers.

**Multicast**: Minder verkeer op het netwerk dan broadcast omdat we enkel naar bepaalde hosts sturen. In IPv4 zijn de adressen 224.0.0.0 tot 239.255.255.255 gereserveerd voor multicast. 224.0.0.0 tot 224.0.0.255 zijn gereserveerd voor multicast op een lokaal netwerk. De hosts moeten ‘subscribed’ zijn aan een bepaalde multicast groep om de packetten ervan te krijgen.

7.1.4: Types of IPv4 Addresses

Private and public IPv4 Addresses: Public IPv4 adressen zijn gerouted door ISP-routers, maar niet alle IPv4 adressen kunnen op het internet worden gebruikt. Private adressen worden door veel organisaties gebruikt, maar kunnen niet op het internet gebruikt worden. Ze zijn niet uniek. De routers gaan de adressen moeten vertalen naar publieke IPv4 adressen door NAT (Network Address Translation). Bijvoorbeeld, thuis heeft elk computer een privéadres, uiteindelijk krijgt heel onze netwerk één public adres. Dit is nodig door het gebrek aan meer IPv4 adressen.

**Privé addressen**: 10.0.0.0 /8, 172.16.0.0 /12, 192.168.0.0 /16 (van 192.168.0.0 tot 192.168.255.255).

Speciale IPv4 Adressen:

**Loopback Adressen**: 127.0.0.0 – 127.255.255.254, gebruikt door een host om verkeer naar zichzelf te leiden. (Kan gebruikt worden om de TCP/IP-configuratie te testen, met een ping).

**Link-Local Adressen:** 169.254.0.1 – 169.254.255.254 🡪 APIPA, Automatic Private IP Adressing, gebruikt door de Windows DHCP client als er geen DHCP-servers beschikbaar zijn.

**TEST-NET adressen**: 192.0.2.0 – 192.0.2.255, gereserveerd voor educatieve doeleinden.

Legacy Classful Addressing

Unicast IPv4 adressen werden verdeeld in verschillende groepen of classes.

**Class A**: 0.0.0.0/8 – 127.0.0.0/8, ontworpen voor zeer grote netwerken met meer dan 16 miljoen hosts. /8 🡪 Eerste octet = Network adres, 3 laatste octetten 🡪 Hosts adressen. Er zijn 128 mogelijke class A-netwerken.

**Class B**: 128.0.0.0/16 – 191.255.0.0, ontworpen voor middelmatige en grote netwerken. Meer dan 65.000 hosts addressen (256\*256-2). /16 🡪 2 octetten voor Network en 2 voor de Host Portion.

**Class C**: 192.0.0.0/24 – 223.255.255.0/24, ontworpen voor kleine netwerken, maximum 254 hosts, enkel laatste octet voor hosts. Meer dan 2 miljoen mogelijke netwerken.

There is also a Class D multicast block consisting of 224.0.0.0 to 239.0.0.0 and a Class E experimental address block consisting of 240.0.0.0 – 255.0.0.0.

Classless addressing: Als we enkel zouden werken met classful addressing, zouden we veel IP-adressen

Verspillen, een bedrijf dat 260 hosts nodig geeft zou een class B nodig hebben, gemaakt voor 65000 hosts. Classful addressing werd om deze reden verlaten (er zijn ook enkel 4 miljard IPv4 adressen).

Vandaag beheren verschillende bedrijven op elk continent de IPv4 en IPv6 adressen. Regional Internet Registries geven de IPv4 addressen aan de ISP en de ISP kunnen ze dan toevertrouwen aan kleinere ISP’s.

7.2: IPv6 Network Addresses:

Migratietechnieken van IPv4 naar IPv6: **Dual Stack**: Het is mogelijk om op één netwerk IPv4 en IPv6 te gebruiken. **Tunneling**: Een methode waar we een IPv6 packet door een IPv4 netwerk transporteren. De IPv6 packet wordt dan geëncapsuleerd in een IPv4 packet. **Translation**: Dmv van NAT64 is het mogelijk voor IPv6 apparaten te communiceren met IPv4 apparaten. IPv6 wordt vertaald in een IPv4 packet en omgekeerd.

Een IPv6 adres bestaat uit 8 hextetten (128 bits) van hexadecimale getallen, in totaal 32 hexadecimale getallen. Vb: x:x:x:x:x:x:x:x, waar elke x staat voor 4 hexadecimale getallen (0-F).

Er bestaan enkele regels om de notatie van een IPv6 adres te versimpelen:

-We laten nullen vooraan een hextet weg. Vb: 0200 wordt 200, 01AB wordt 1AB.

-Als er een segment bestaat met enkel nullen, laten we deze weg, maar schrijven we wel de dubbelpunt.

De dubbele dubbelpunt kan enkel één keer gebruikt worden in een adres.

Voorbeeld: 2001:0DB8:0000:1111:0000:0000:0000:0200 🡪 2001:DB8:0:1111::200

Voorbeeld: 0000:0000:0000:0000:0000:0000:0000:0001 🡪 ::1 (Loopback adres).

IPv6 Address Types: Naast Unicast en multicast bestaat er met IPv6 ook **Anycast**. Een IPv6 anycast adres is een IPv6 unicast adres dat aan meerdere apparaten kan worden gegeven. Het packet wordt dan gerouted naar de dichtstbijzijnde apparaat met dat adres. Broadcast bestaat niet met IPv6, maar er bestaat wel een all-nodes multicast adres, dat hetzelfde doet. (FF02::1)

IPv6 prefix length: Het prefix kan van 0 tot 128 groot zijn bij IPv6 adressen. Het duidt de Network Portion aan van een IPv6 adres. Een typische prefix lengte voor een LAN is /64, dit betekend dat de Interface ID (Host Portion) ook 64 bits lang is.

IPv6 Unicast adressen: De meest voorkomende IPv6 unicast adressen zijn global unicast adressen en link-local unicast adressen.

**Global Unicast:** Zoals een public IPv4 adres, ze zijn wereldwijd uniek.

**Link-Local:** Worden gebruikt om te communiceren met andere apparaten op dezelfde link (subnet), packets van link-local adressen wordt niet gerouted. Range: FE80::/10.

**Unique Local:** Gebruikt voor local addressing op een gebied of tussen gebieden, maar kunnen niet gerouted worden tot een globaal IPv6-adres. Range: FC00::/7 to FDFF::/7.

Structuur van een IPv6 Global Unicast Adres: Momenteel worden er enkel IPv6 adressen met als 3 eerste bits 001 of 2000::/3 toegewezen, dit is 1/8ste van alle mogelijke IPv6 adressen. Een global unicast adres bestaat uit 3 delen, **Global Routing Prefix**, **Subnet ID**, **Interface ID.**

**Global Routing Prefix:** Vaak /48 prefix, met de 3 eerste hextetten als network portion. De lengte van de global routing prefix bepaalt ook de lengte van de subnet ID. (/48 routing prefix + 16 bits subnet ID = /64 prefix).

**Subnet ID:** Wordt gebruikt om de subnets in een organisatie te identificeren.

**Interface ID:** Vergelijkbaar met de Host Portion van een IPv4 adres.

Static configuration of a Global Unicast Address: De meeste IPv6 configuratie commandos in de Cisco IOS zijn gelijk aan die van IPv4; soms enkel ipv6 schrijven ipv ip. De commando om een global unicast address op een interface te configureren is: **ipv6 address** *ipv6-address/prefix-length.* Niet vergeten: **no shutdown**.

Een apparaat kan dmv DHCPv6 of SLAAC (Stateless Address Autoconfiguration) automatisch een IPv6 global unicast adres krijgen.

IPv6 routing is not enabled by default. To enable a router as an IPv6 router, the **ipv6 unicast-routing** global configuration command must be used.

Alle IPv6 apparaten moeten een IPv6 Link-Local adres hebben, deze kan dynamisch gegeven worden, of statisch geconfigureerd. Om een Link-Local adres te configureren op een router gebruiken we dezelfde werkwijze als hierboven maar schrijven we link-local erachter: **ipv6 address** *ipv6-address* **link-local**.

Verifying IPv6 Address Configuration: **show ipv6 interface brief**; **show ipv6 route**; **ping**.

IPv6 multicast adressen zijn gelijkaardig aan IPv4 multicast adressen. IPv6 Multicast adressen hebben als prefix **FF00::/8**. Er bestaan twee soorten IPv6 multicast adressen **assigned** en **solicited node multicast**.

**Assigned Multicast Adressen:** Zijn adressen die gereserveerd zijn voor specifieke apparaten. Vb: All nodes multicast groep: Een multicast voor alle IPv6 devices; All-routers multicast groep: Een groep dat alle IPv6 routers kunnen joinen.

**Solicited-Node IPv6 Multicast Adressen:** Gelijkaardig aan de all-node multicast adres.

7.3.1: ICMP (Internet Control Message Protocol)

ICMP bestaat zowel voor IPv4 as voor IPv6. Deze berichten geven feedback over problemen met IP packets. Belangrijke ICMP berichten zijn: Host confirmation; Destination or Service Unreachable; Time exceeded; Route redirection.

4 nieuwe berichten in ICMPv6 met de Neighbor Discovery Protocol: **Router Solicitation** (RS, toestel naar router), **Router Advertisement** (RA, router naar toestellen om de zoveel tijd). RA wordt gebruikt om adres informatie te sturen naar hosts die SLAAC gebruiken.

Tussen apparaten: **Neighbor Solicitation message** (NS, gebruikt wanneer een apparaat het IPv6 adres van een ander apparaat kent, maar niet het MAC-adres, zoals een ARP request voor IPv4). Kan ook gebruikt worden om te zien of iemand anders dezelfde IP-adres heeft. Geen antwoord op netwerk = uniek IP-adres.

**Neighbor Advertisement message** (NA, antwoord op NS, zoals ARP-reply in IPv4).

7.3.2: Testing and verification

Een ping is een connectivity test dat ICMP gebruikt. Met een ping sturen we een echo request, als de host de echo request krijgt, stuurt het een echo reply.

Connectivity to local LAN testen: We gaan dan vaak pingen naar de default gateway, maar we kunnen ook gewoon pingen naar een andere host op het netwerk. Als we een antwoord krijgen dan kunnen we communiceren op de lokale netwerk. Als de gateway niet antwoord maar een andere host wel, dan is er waarschijnlijk een probleem met de router interface. Een slechte gateway adres op de host bijvoorbeeld.

Ping kan ook gebruikt worden om te zien of een host kan communiceren op het internet. We kunnen dan een host pingen op een ander netwerk. Als dit werkt weten we dat de host kan communiceren op het netwerk en dat alle routers op de weg juist geconfigureerd zijn.

Traceroute (tracert) wordt gebruikt om de details van de route van een bericht te kennen. We kunnen zo een lijst opstellen van alle routers op de weg. Als de data ergens een probleem tegenkomt kunnen we weten wat de laatste succesvolle router was op de weg en waar het probleem zich bevindt.

**Round Trip Time** (RTT): De tijd dat een packet nodig heeft om tot een remote host te gaan, en de tijd nodig voor een reply.

IPv4 TTL en IPv6 Hop Limit: Traceroute gebruikt TTL en Hop Limit om informatie te krijgen over de routers op de weg. Het begint met een TTL van 1 en elke keer dat het een timeout message krijgt incrementeerd de TTL, totdat het bericht tot bij de host komt. (Of dat het een maximum TTL bereikt). Elke keer dat het bericht timeout, moet de router een ICMPv4 message terugsturen. Hiermee krijgt tracert het adres van de router.

# Hoofdstuk 8: Subnetting IP Networks

8.1.1: Network Segmentation

In een ethernet LAN wordt broadcast gebruikt om andere apparaten (ARP-broadcast) en diensten te lokaliseren (DHCP). Switchen verspreiden de broadcasts in alle richtingen (behalve de poort waardoor het de broadcast krijgt), routers verspreiden broadcast niet. Op een groot netwerk met veel hosts kunnen deze excessieve broadcasts voor problemen zorgen. Ze zorgen dan voor te veel verkeer en vertragen het netwerk, de apparaten gaan ook trager werken omdat ze constant broadcasts binnen krijgen en verwerken. Een oplossing is om het netwerk te subnetten (om kleinere netwerken te maken op één netwerk). De kleinere netwerken noemen we subnets.

Andere redenen om te subnetten zijn: per locatie, per soort apparaat, per soort werknemer, …

8.1.2: Subnetting an IPv4 Network

IPV4 subnets worden gemaakt door een of meer host bits te gebruiken (lenen) als network bits om zo meer netwerken te kunnen maken met minder hosts per netwerk. Makkelijkst om te subnetten zijn de octet boundaries: /8, /16, /24 (Reminder: dit stelt de lengte van de prefix (network portion) voor).

Classless subnetting: subnetten, maar niet met de octet boundaries. Vb: /25, /18, /12, …

**Hoeveel subnets?** Voor elk bit dat we gaan lenen creëren we een nieuw aantal subnets (netwerken): 2n. Vb: We lenen 1 bit: 21 = 2 subnets. (We noemen de geleende bits subnet bits).

**Hoeveel hosts kunnen er op elk subnet?** We tellen eerst de host bits. Daarna 2n-2. (2n min de broadcast adres en de default gateway).

Vb: /25 🡪 7 host bits dus: 27-2 = 126 hosts per netwerk.

Stel: we lenen één bit om een /25 adres te maken, subnet mask wordt 255.255.255.128, als het IP-adres 192.168.1.0/24 was, dan hebben we nu twee netwerken: 192.168.1.0 /25 tot 192.168.1.127 /25 en 192.168.1.128 /25 tot 192.168.1.255 /25. Met op elk netwerk 126 mogelijke hosts. (Zie ook magic number technique).

/30 netwerk is het maximum, anders niet meer genoeg adressen voor hosts. (/30 = 2²-2 hosts).

Zie video 8.1.2.4

De eerste host adres is altijd 1 meer dan het netwerk adres, de laatste 1 minder dan de broadcast adres (allemaal 1-en in host portion behalve laatste).

Zie video 8.1.3.6 – subnetting across multiple octets. Naast de magic number in laatste octet waar we een bit lenen (die we gaan incrementeren met de magic number), gaan we ook de octetten ervoor waaruit we bits lenen moeten laten toenemen met 1 tot 255.

Wanneer we willen gaan subnetten, moeten we met 2 zaken rekening houden: - Hoeveel host adressen we willen voor elk netwerk en hoeveel verschillende netwerken we nodig hebben. Deze twee waarden zijn omgekeerd evenredig. We moeten ook letten op het feit dat er twee adressen niet door hosts gebruikt kunnen worden omdat ze gebruikt worden voor ons netwerkadres en broadcast adres (2^n-2).

8.1.5: Benefits of Variable Length Subnet Masking

Met de traditionele manier van subnetten bestaan er gevallen waar we veel adressen niet gaan gebruiken.

VLSM of Variable Length Subnet Masking werd uitgevonden om te vermeiden dat we adressen gaan verspillen.

Bij traditionele subnetting geven we dezelfde subnet mask aan alle adressen, waardoor ze allemaal evenveel bruikbare hosts hadden per netwerk. VLSM kan een subnet verder verdelen in verschillende delen. Dit gebeurt op een dezelfde manier dat we hiervoor hebben gezien. We gaan eerst een netwerk subnetten, en de subnets gaan we nog eens subnetten. De regels om het aantal subnets en hosts per subnet te bepalen blijven ook hetzelfde.

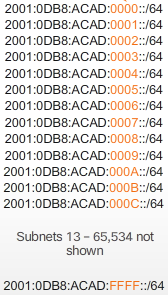
Zie video: 8.1.5.4 – Basic VLSM + 8.1.5.7 – VLSM Example

8.2.1: Structured design

Voor we beginnen te subnetten moeten we eerst een network requirement study maken, om te zien wat een bedrijf nodig heeft. We moeten kijken naar hoeveel verschillende subnets we nodig hebben, hoeveel verschillende hosts per subnet en hoe we de adressen aan elk host gaan geven (welke hosts hebben een statisch adres nodig? Welke kunnen DHCP gebruiken om een adres te krijgen?)

8.3.1: Subnetting an IPv6 Network

In IPv4 gaan we subnetten om broadcasts te limiteren en om de schaarste van IPv4 adressen tegen te gaan, bij IPv6 adressen moeten we ons geen zorgen maken over te weinig adressen. Bij IPv6 adressen gaan we subnetten om een hiërarchie te maken gebaseerd op het aantal subnetwerken die we nodig hebben. IPv6 Link-Local adressen gaan we nooit subnetten omdat het enkel op de lokale link bestaat, maar IPv6 global unicast adressen kunnen we wel subnetten. De IPv6 global unicast adres heeft vaak een /48 global routing prefix gevolgd door een 16 bit subnet ID en een 64 bit interface ID.

De 16 bits Subnet ID kan gebruikt worden door een organisatie om interne subnets te maken. We hebben dan meer subnets en hosts support dan we ooit nodig zullen hebben. We kunnen tot 65.536 /64 subnets maken (zonder bits te lenen van de interface ID) en tot 18 quintillion host IPv6 adressen per subnet hebben.

IPv6 subnetting is gemakkelijker dan IPv4, omdat we niets naar het binair moeten converteren, we moeten enkel tellen in hexadecimaal.

# Hoofdstuk 9 : Transport Layer

9.1.1 : Transportation of Data

De transport layer dient om een tijdelijke communicatiesessie op te richten tussen twee applicaties en data te bezorgen tussen de applicaties. Een applicatie van de source host produceert data dat gestuurd wordt naar de applicatie van de destination host. De Transport Layer is een link tussen de application layer en de lagen eronder die zorgen voor ‘network transmission’ (TCP/IP Model).

Op het transport laag is elke ‘set’ van data tussen een source en destination applicatie gezien als een conversatie. Een host kan meerdere applicaties tegelijk gebruiken die elk met een of meer applicaties communiceren met een remote host. Het is dan de rol van het transport laag om deze verschillende gesprekken te onderhouden en op te sporen. De meeste netwerken hebben beperkingen in hoe groot een packet kan zijn. De transport layer zorgt er ook voor om de application data te segmenteren in verschillende blokken van geschikte grootte (inclusief de incapsulatie voor elk deel van het data). Eens de packets tot hun bestemming komen moet het transport laag ze ook weer reconstrueren. De transport layer header wordt gebruikt om ze weer in de goede volgorde te plaatsen. Om te weten welke packetten tot welke applicatie behoren, krijgt elke applicatie een ‘port number’ (een ID).

De reden waarom we de data segmenteren in kleinere stukken is dat het zo makkelijker is om fouten te vinden en te herstellen. Een complete communicatie stream (vb. video) zou ook heel het bandbreedte kunnen gebruiken, waardoor we geen verschillende communicaties tegelijk zouden kunnen voeren met verschillende applicaties. Elke segment van data wordt geïdentificeerd d.m.v. datavelden in de header (bij de encapsulatie).

IP is niet bezig met hoe een packet geleverd moet worden op het netwerk. Daarvoor gebruiken we andere protocollen, Transport protocollen. Deze specificeren hoe een packet moet worden overgebracht tussen de hosts. TCP/IP heeft 2 transport layer protocollen. **TCP** (Transport Control Protocol) en **UDP** (User Datagram Protocol).

**TCP**: Nummert en stuurt data in verschillende delen, wacht op een bevestiging dat een deel gekregen werd voor het de volgende deel stuurt. Krijgt het geen bevestiging voor een bepaald deel, dan stuurt het dat deel opnieuw. TCP is betrouwbaar. (Mail, http).

**UDP**: Stuurt alle data in één keer, wacht niet op een bevestiging, daardoor is het veel sneller maar minder betrouwbaar (best-effort). UDP is wel nodig voor bepaalde applicaties, (VoIP, Games, live video).

9.1.2: TCP and UDP Overview

TCP:

**Establishing a session**: Verzekeren dat de applicatie klaar is om data te ontvangen.

**Reliable Delivery**: Alle segmenten van de data worden geleverd en zijn compleet.

**Same order delivery**: Verschillende packets kunnen verschillende wegen nemen, uiteindelijk worden ze in de goede volgorde geassembleerd.

**Flow Control**: Verzekert dat de ontvanger de data krijgt en kan processen.

De TCP-header: Bestaat uit 20 bytes. Hierin staat de **source en destination port** om de applicatie te identificeren, de **sequence number** om ze later te re-assembleren in de juiste volgorde, de **acknowledgment number** om te weten welke data al verkregen werd en de **checksum** (error checking van header en data).

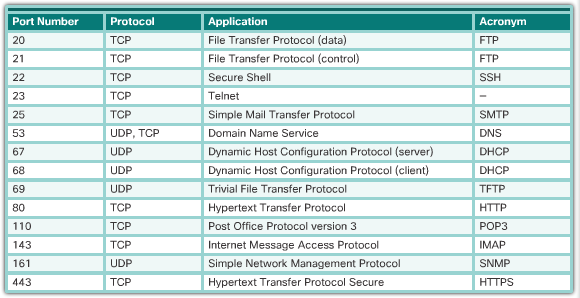
UDP:

Geen session, geen flow control, geen reconstructie van data in juiste volgorde, niet betrouwbaar.

De UDP-header bestaat uit de **source en destination port**, de **lengte van de packet** en de **checksum**.

De source port wordt dynamisch gegenereerd door de zender, de destination port vertelt de server welke dienst gebruikt moet worden, (bv. Poort 80 = HTTP).

Port nummers: van 0 tot 1023: Welgekende poorten, zijn gereserveerd voor diensten en applicaties (webbrowsers, email clients…). Van 1024 tot 49151: geregistreerde poorten, bijvoorbeeld voor individuele applicaties die minder ordinair zijn. Van 49152 tot 65535, dynamische of private-ports. Worden dynamisch gegeven door de OS wanneer een connectie gemaakt wordt om de applicatie te identificeren gedurende de communicatie.

Te kennen poorten en protocollen:

Om te weten welke actieve TCP-connecties open zijn kunnen we in cmd de commando **netstat** gebruiken.

9.2.1: TCP Communication process

Elke applicatie is geconfigureerd om een poort nummer te gebruiken. Een server kan geen verschillende diensten toewijzen aan éénzelfde poort.

TCP connection establishment, 3 stappen: (SYN ACK)

1. De startende client vraagt een client-to-server communicatie sessie met de server (SYN, Synchronization).

2. De server bevestigd de client-to-server communicatie sessie en vraagt een server-to-client communicatie sessie (SYN ACK, Acknowledgment).

3. De startende client bevestigd de server-to-client communicatie sessie (ACK).

TCP Connection termination (FIN ACK)

1. Wanneer de client geen data te versturen meer heeft, stuurt het een segment met de FIN flag. (FIN)

2. De server stuurt een ACK terug om te tonen dat hij de FIN heeft gekregen om de client-to server sessie te stoppen. (ACK)

3. De server stuurt een FIN om de server-to-client sessie te stoppen. (FIN)

4. De client stuurt een ACK terug om te tonen dat het de FIN heeft gekregen. (ACK)

Three-way handshake:

1. Establishes that the destination device is present on the network.

2. Verifies that the destination device has an active service and is accepting requests on the destination port number that the initiating client intends to use.

3. Informs the destination device that the source client intends to establish a communication session on that port number.

De 6 control bits in de TCP-segment header worden ook flags genoemd (URG, ACK, PSH, RST, SYN, FIN).

9.2.2: Reliability and Flow Control

Om ervoor te zorgen dat alle segmenten in de juiste volgorde worde geplaatst na dat ze geleverd werden, wordt er een Initial Sequence Number (ISN) gegeven. De ISN stelt de startwaarde van de bytes voor dat sessie voor. Het incrementeerd door het aantal bytes dat worden verzonden. Als er segmenten worden gemist, kunnen we dat meteen zien.

Video 9.2.2.2 – Sequence Numbers and Acknowledgments. + 9.2.2.3 Data Loss and Retransmission.

Windows size: Het aantal bytes dat een apparaat tegelijk kan krijgen en processen. De window size wordt gekozen tijdens de three-way handshake. Dit is een mechanisme voor de flow control, het zorgt ervoor dat de TCP-transmissie betrouwbaar blijft. De window size staat in de TCP-header.

Congestion avoidance: Wanneer er opstopping is op het netwerk, worden packets door de overladen router gedropt. Ze worden dan ook nooit een bevestiging. Door het tempo waarop packets worden gestuurd maar niet bevestigd worden vast te stellen, kan de source weten hoeveel ophoping er op het netwerk is. De source kan dan kleinere aantallen bytes doorsturen voor het een acknowledgement krijgt. (Pas op: de windows size blijft wel onveranderd).

9.2.3: UDP Communication

Datagrams worden niet geordend als ze in de slechte volgorde aankomen, als ze niet aankomen worden ze ook niet meer verstuurd. Voor applicaties waarvoor dat belangrijk is kunnen de applicaties dat zelf doen.

9.2.4: TCP or UDP

TCP: HTTP, FTP, SMTP (simple mail transfer protocol), Telnet.

UDP: DHCP, DNS, SNMP (simple network management protocol), TFTP, VoIP, IPTV

3 Soorten applicaties die het best met UDP gebruikt worden: Live video/multimedia applicaties, simpele requests en reply applicaties (DNS, DHCP). Applicaties die zelf zorgen voor de betrouwbaarheid (SNMP, TFTP).

# Hoofdstuk 10 : Application Layer

10.1.1: Application, Presentation and Session

De Application Layer is het dichts bij de gebruiker, het is de 7de laag van het OSI-Model en de 4de laag van het TCP/IP Model. In de TCP/IP Model staan ook de **Presentation** en **Session** laag van het OSI-model in de Application Layer. Het is de laag dat een interface levert tussen de applicaties om te communiceren en het onderliggend netwerk waarover berichten worden verzonden. De protocols van de application layer worden gebuikt om data om te wisselen tussen de programma’s op de source host en de destination host.

Er bestaan veel Application Layer Protocols, zoals HTTP, FTP, TFTP, IMAP, DNS, DHCP, POP, SMTP…

**Presentation Layer**: Heeft 3 belangrijke functies. (Formateren, compresseren en encrypteren).

1. Formatering (of presenteren) van data op het source apparaat tot een compatibele vorm voor de ontvangst ervan op de destination apparaat. (JPEG, PNG, MP4…)

2. Compresseren van de data op een manier waarop het kan worden gedecompreseerd op de destination apparaat.

3. Data Encrypteren om het te verzenden en decrypteren eens het verkregen werd.

**Session Layer**: Regelt de totstandkoming, onderhoudt en beëindigt een sessie tussen twee communicerende hosts

TCP/IP Application Layer Protocols: DNS, BOOTP, DHCP, SMTP, POP, IMAP, FTP, TFTP, HTTP, HTTPS.

10.1.1.3 voor meer informatie over elk protocol.

10.1.2: How Application Protocols Interact with End-User Applications

Client-Server Model: Het toestel dat informatie aanvraagt is een client, het apparaat dat antwoord op de aanvraag is een server. De client en server processen zijn in de application layer. Client tot server = upload, server tot client = download.

Peer-to-Peer netwerk: In P2P networking krijgt een peer toegang tot data door een andere peer zonder het gebruik van een server. In een P2P netwerk zijn 2 of meer computers aan elkaar gelinkt via een netwerk en kunnen ze resources met elkaar delen zonder gebruik te maken van een server. Bijvoorbeeld PC1 (peer1) print een document uit dat op de harde schijf staat van PC2 (peer2) over het netwerk.

Een P2P applicatie maakt het mogelijk voor apparaten te fungeren als client en server in éénzelfde communicatie. In deze model is elke client een server, en elke server een client. (Vb: BitTorrent, Bitcoin).

10.2.1: Web and Email Protocols

HTTP en HTML: Als we een adres zoeken, wordt het adres omgezet in een numeriek adres, dan gaat er een request naar een http-server (met numeriek adres), de browser stuurt een GET request voor de index.html. De server stuurt dan de HTML-code naar de browser.

HTTP is niet veilig, de informatie wordt niet geëncrypteerd, de requests worden gestuurd naar de server in “plain text”. Voor veilige communicatie wordt HTTPS (secure) gebruikt. Het gebruikt encryptie en authenticatie om data te beveiligen tussen de client en server. (Encryptie met SSL).

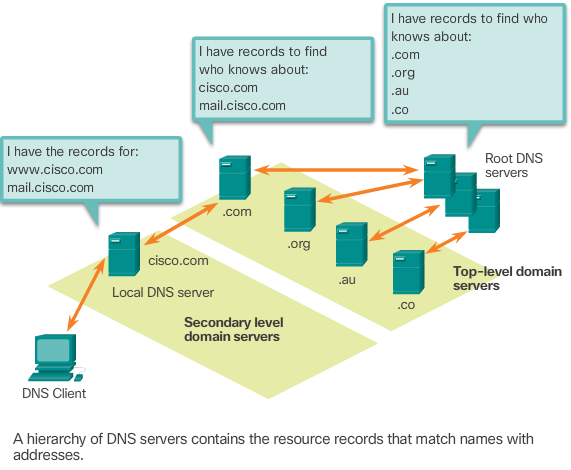
Email Protocollen: Email clients communiceren niet met elkaar, maar communiceren met Mail Servers (die tussen elkaar communiceren). Er bestaan 3 email protocollen: SMTP (Simple Mail Transfer Protocol), POP (Post Office Protocol) en IMAP (Interactive Message Access Protocol). SMTP wordt gebruikt bij het versturen, POP (mail op computer) en IMAP (mail op server) bij het verkrijgen van de mail.

SMTP: Een SMTP bericht moet een message header en message body hebben. De header moet het adres van de zender bevatten en dat van de ontvanger. Wanneer een mail wordt gestuurd verbindt de client SMTP-proces met een server SMTP process op poort 25. Na de verbinding probeert de client de email te sturen naar de server. De server krijgt de mail en stuurt het naar de ontvanger als deze een local is. Anders stuurt de server de mail door naar een andere server.

POP: Gebruikt door een applicatie om een mail van de server te halen. Het wordt gedownload naar de client en dan verwijderd van de server. De server start de POP-service door te luisteren naar poort 110. Wanneer de client de service wilt gebruiken, stuurt het een request om een TCP connectie te maken met de server.

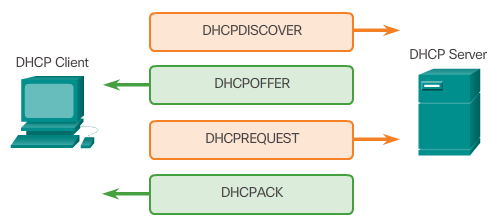
IMAP: Kopieën van de mail worden gedownload naar de client applicatie, ze blijven op de server tot ze manueel verwijderd worden.

10.2.2: IP Addressing Services

DNS: Domeinnaam (vb. cisco.com) omzetten naar IP-adres. Als IP-adres veranderd, blijft cisco.com dezelfde. Eens de computer het IP-adres heeft, kan het communiceren met de server. Als een DNS-server geen match vindt voor een naam, contacteert het andere DNS servers, eens een match op een andere server is gevonden, zal de eerste server die tijdelijk opslaan in het geval iemand het opnieuw vraagt. **Ipconfig /displaydns** om de cached DNS-entries van de ‘DNS Client Service’ op windows te zien.

DNS-hiërarchie: De DNS-servers hebben een hiërarchie om de namen te vinden, het lijkt op een omgekeerde boom, met de wortels boven en de takken beneden. De structuur van de naam van een website wordt verdeeld in meerdere delen, elke DNS-server heeft een specifieke database.

- Commando **nslookup** in cmd om de IP-adres van een domeinnaam te krijgen.

DHCP: Dynamic Host Configuration Protocol. Een apparaat wordt opgestart, de client stuurt een DHCPDISCOVER in broadcast om een DHCP-server te vinden. Een DHCP-server antwoord met DHCPOFFER bericht, om een ‘lease’ aan te bieden, het bericht bevat een IPv4-adres en een subnet mask dat toegewezen moet worden, de IPv4-adres van de DNS-server en de IPv4-adres van de default gateway. De HDCPOFFER zegt ook hoe lang de lease duurt.

Een client kan meerdere DHCPOFFER berichten krijgen, dus antwoord het met een DHCPREQUEST bericht naar de server waarvan de client de lease accepteert. De server retourneert een DHCPACK (acknowledgment) bericht om te zeggen dat de lease gemaakt werd. Als de lease niet meer geldig is, stuurt de server een DHCPNAK (negative ack). Dan moet de client opnieuw beginnen.

DHCPv6 has similar set of messages to those shown in the figure for DHCP for IPv4. The DHCPv6 messages are SOLICIT, ADVERTISE, INFORMATION REQUEST, and REPLY

FTP: Werd gemaakt voor data transfer tussen client en servers. Er worden twee connecties gemaakt om data te transfereren met FTP.

1. De client maakt een connectie met server voor control traffic (TCP port 21), bestaande uit client commando’s en server reply’s.

2. De Client maakt een tweede connectie met de server voor de eigenlijke data transfer (met TCP port 20). Elke keer dat er data wordt getransfereerd wordt deze connectie opnieuw gemaakt.

Server Message Block (SMB): Client/Server File sharing protocol op windows. Advantage over FTP: SMB clients can establish a long-term connection to the server. ​

# Hoofdstuk 11: Build a Small Network

11.1.1: Devices in a Small Network

Wanneer we een klein netwerk maken, moeten we de IP addressing space plannen. Bijvoorbeeld alle servers dezelfde logical block van adressen toekennen in het adresbereik van het netwerk.

Ook redundancy (met switches, routers en servers) is belangrijk op een klein netwerk. Om ervoor te zorgen dat het netwerk blijft werken als er een probleem is met een netwerktoestel.

We kunnen prioriteiten geven aan het verkeer op de router. Bijvoorbeeld VoIP krijgt hogere prioriteit dan SMTP of Webpagina’s.

11.1.2: Small Network Applications and Protocols

Elke applicatie of network service gebruikt protocollen, ze bepalen de standaarden en formaten die gebruikt moeten worden. Zonder protocollen zouden we geen algemene manier hebben om data te besturen. (Goed kennen 🡪 DNS Server, Telnet, email server, DHCP-server, webserver, ftp-server).

11.1.3: Scale to Larger Networks

Om te weten welke soort verkeer er veel wordt gebruikt om een netwerk kunnen we een protocol analyzer gebruiken.

11.2.1: Security Threats and Vulnerabilities

Diefstal van informatie: Inbraak in computer om confidentiële informatie te stelen.

Verlies van informatie: Inbraak in computer om data te vernielen of te wijzigen.

Identiteitsdiefstal: Persoonlijke informatie stelen om de identiteit van iemand te stelen.

Onderbreking van dienst: Ervoor zorgen dat mensen geen toegang hebben tot diensten (DoS).

De fysieke veiligheid van toestellen in ook heel belangrijk. Tegen fysieke schade van hardware, of schade door milieu (te warm/koud), of elektriciteit (voltage spikes).

Er bestaan 3 soorten kwetsbaarheden: Technologische, configuratie en veiligheidsbeleid kwetbaarheden.

11.2.2: Network Attacks

Soorten Malware: Virussen, wormen en Trojaanse paarden.

**Virus**: Een programma dat zich in een bestand of andere programma kan nestelen.

**Worm**: Zelfstandige programma’s die zich over het netwerk verspreiden. (Geen programma nodig om zich aan vast te hechten).

**Trojan Horse**: Een functie verborgen in een ander programma dat door de gebruiker wordt geïnstalleerd.

Soorten netwerkaanvallen:

**Reconnaissance attacks**: Vinden en in kaart brengen van systeem, diensten of kwetsbaarheden. (Ping, whois, packet sniffers…)

**Access attacks**: Manipulatie van data, toegang tot het systeem of gebruikersrechten. (Paswoorden vinden, Man-in-the-Middle, port redirection…)

**Denial of Service**: Een computer, netwerk of dienst onbeschikbaar te maken voor de bedoelde gebruiker.

11.2.3: Network Attack Mitigation

Authentication, authorization, and accounting (AAA) 🡪 Controle over wie op het network kan (Authentification), wat ze kunnen doen op het network (authorization) en welke acties ze uitvoeren terwijl ze op het netwerk zijn (accounting). (Zoals creditcard, code, hoeveel kan je uitgeven? Waarvoor betaald?)

Firewall: Netwerk firewalls staan tussen twee of meer netwerken en beveiligen tegen ongeoorloofde toegang. Dat kan op basis van IP/MAC-adres, poort nummer, URL, sleutelwoorden of Stateful Packet Inspection (**SPI**) waarbij inkomende packets antwoorden moeten zijn op requests van hosts binnenin. Ongevraagde packetten worden dan geblokkeerd.

11.2.4: Device Security

Standaard paswoorden en usernames veranderen op nieuwe switches en routers. Sterke paswoorden gebruiken.

Commando’s voor paswoorden in IOS: **security password min-length** ***x*** 🡪 paswoorden moeten minimum x karakters lang zijn. **Login block-for *120* attemps *3*  within *60*** 🡪 Blokkeert voor 120 seconden als mensen 3 keer een fout wachtwoord proberen in 60 seconden. (Global)

**Exec-timeout 10** 🡪 Sluit gebruiker uit van lijn na 10 minuten inactiviteit. (Console, VTY, aux poorten).

SSH activeren op apparaten: 11.2.4.4 + Packet tracer activity.

Step 1: Configure the IP domain name.

Step 2: Generate one-way secret keys.

Step 3: Verify or create a local database entry.

Step 4: Enable VTY inbound SSH sessions.

11.2.5: Backup and Restore Configuration Files

Navigeren in de bestanden van een Cisco Router (IFS, IOS Files System):

**show file systems** 🡪 Lijst van alle beschikbare bestanden op de router.

**dir** 🡪 Lijst van inhoud van de default file system, bijvoorbeeld flash.

Om de inhoud van NVRAM te zien, moeten we de default file system veranderen. We gebruiken **cd nvram:** daarna **pwd** (present working directory) om te controleren dat we wel in de NVRAM zijn. Daarna **dir**.

Backup in .txt met Tera Term 🡪 11.2.5.3

Backup met TFTP:

Step 1. Enter the **copy running-config tftp** command.

Step 2. Enter the IP address of the host where the configuration file will be stored.

Step 3. Enter the name to assign to the configuration file.

Step 4. Press Enter to confirm each choice.

Restore is hetzelfde maar met commande **copy tftp running-config**.

11.3.1.1: Interpreting Ping Results

Op IOS zijn er verschillende indicatoren voor pings (ICMP echo requests) de belangrijkste zijn: **!** 🡪 Toont aan dat de ping gekregen werd. **. (punt)** 🡪 Toont aan de tijd verlopen is om te wachten op een reply.

**U** 🡪 Een ICMP unreachable bericht werd verkregen. (Door een router, ping request geblokkeerd, of de router heeft de weg tot de bestemming niet).

11.3.2: The traceroute and tracert Command

Op windows: **tracert** op IOS: **traceroute**

11.3.3: Show Commands

Belangrijke show commando’s:

show running-config

show interfaces

show arp

show ip route

show protocols

show version

11.3.4: Host and IOS Commands

Arp commando: **arp -a** (windows) 🡪 Alle toestellen in de arp cache van de host.

Cisco Discovery Protocol (CDP) 🡪 Om aangrenzende Cisco toestellen te vinden. 🡪 **show cdp neighbors**. Om CDP te deactiveren gebruiken we  **no cdp enable**.

**show ip interface brief** 🡪 Een samenvatting van de informatie van alle netwerk interfaces op een router of switch. Er bestaat ook show ip interface, dat is dan alle informatie.