Lagen
- Physical 1
- Datalink 2
- Network 3
- Transport 4
- Session 5
- Presentation 6
- Application 7

Fysieke laag protocollen
- WAP: draadloos toeganspunt (verbinden met netwerk)
- ISR: integrated service routers (mogelijkheid tot kabel als draadloos verbinden met LAN)
- NIC: network interface card
- Doel: bits van frame, gevormd door datalinklaag, over het netwerk medium te sturen
- Copper cable: elektrische signalen
- Fiber-optic cable: light pulse
- Wireless: microwave signals
- Governed by: ISO, EIA/TIA, ITU-T, ANSI, IEEE

Hoe signalen verzenden
- Asynchroon: data signalen verzonden zonder bijbehorend kloksignaal, tijd tussen data karakters of blokken kunnen van willekeurige duur zijn -> afstand niet gestandaardiseerd -> start en stop indicator vlaggen
- Synchroon: signalen verzonden samen met een kloksignaal -> bitstijd

Bandbreedte
- Bandbreedte: hoeveel data per seconde door een verbinding kan bits/second (bps)
- Kanaalcapaciteit: gebruikte bandbreedte van beschikbare

Throughput
- De hoeveelheid doorgestuurde bytes door een netwerk per tijdseenheid
- Niet hetzelfde als bandbreedte door factoren die invloed hebben: hoeveelheid verkeer, aard verkeer, latentie door netwerkinrichtingen tussen bron en bestemming

Goodput
- Bruikbare data over een bepaalde periode van tijd.
- Goodput = throughput – (overhead voor oprichting sessies + bevestigingen + inkapseling)

Beperkingen koper medium
- Signaaldemping
- Signaalinterferentie: EMI = electromagnetische interferentie, RFI = radio frequency interferenite en/of crosstalk

Soorten koperen kabels en componenten
- UTP: unshielded twisted-pair
- STP: shielded twisted pair
- FTP: foiled twisted pair

- RJ-45 connector
- Coaxial cable
- Coax connector: BNC, N-type of F-type

UTP kabels
- Geen afscherming om EMI en RFI tegen te gaan -> beperken door annulering -> 2 draden met elkaar torsen en de draaiïngen variëren per draadpaar
- Outer jacket: beschermt koperen kabel
- Twisted pair: beschermt tegen signaalinterferentie
- Color-coded plastic insulation: isoleert elke kabel van een ander en indentificeert deze
- TIA/EIA: definieert: kabel-type, kabel-lengte, connectoren, methode om kabel te testen
- IEEE: definieert de elektrische kenmerken van de koperkabel (vb: performantie)
- Category 3: gebruikt voor telefoonlijnen
- Category 5 en 5e: data overdracht van 100mb/s – 1000mb/s
- Category 6: data overdracht van +1000mb/s
- Connectoren: RJ-45 (uiteinde kabel), TIA/EIA 568 (waar je kabel in steekt)
- Ethernet Straight-through: beide uiteinden T568A of T568B -> verbinden network host met network device (vb met: switch, hub)
- Ethernet crossover: één uiteinde T568A, andere T568B -> verbinden 2 network hosts of verbinden 2 intermediary devices (vb: switch met switch, router met router)
- Rollover: cisco patent, verbind workstation serial port met router console port using an adapter
- UTP-kabel tester test: kabel map, kabellengte, signaalverlies door demping, crosstalk

FTP kabels
- UTP maar met foil shield (aluminium bescherming) onder jacket en rond elk twisted pair

Coaxiale kabels
- Outerjacket
- Braided koperen bescherming
- Plastic insulation
- Koperen conductor

Koper medium en veiligheid
- Splitsing tussen data en elektrische stroom kabels moet voldoen aan veiligheids normen.
- Kabels moeten correct verbonden zijn
- Installaties moeten geïnspecteerd worden voor schade
- Equipment must be grounded correctly

Glasvezelkabel
- Gebruikt voor: bedrijfsnetwerken, Fiber-to-the-home (FTTH) en toegansnetwerken, lange afstand netwerken, onderzeese netwerken.
- Langere afstand + hogere bandbreedte
- Minder signaaldemping
- Geen EMI en RFI
- Jacket: PVC omhulsel ter bescherming
- Strenghtening material: omringt buffer zodat glasvezelkabel niet uitrekt als er aan getrokken wordt
- Buffer: bescherming van kern en bekleding

- Cladding: ander soort glas dat kern omspiegelt en fungeert als spiegel dat licht terug naar kern reflecteert
- Core: puur glas (silica), hierin zet het licht zich voor

SMF glasvezelkabel
- Single core
- Less dispersion
- Suited for long distance
- Lasers as light source
- Commonly used with campus backbones for distances of several kilometers

MMF glasvezelkabel
- Larger core
- Greater dispersion <- loss of signal
- Suited for long distance but less than SMF
- Uses LEDs as light source
- Commonly used with LANs or distance of couple 100 meters within a camput network

Glasvezelkabels connectors en cords
- Connectors: ST, SC, LC, duplex multimode LC
- Patch cords: SC-SC multimode, LC-LC single mode, ST-LC multimode, SC-ST single-mode

Testen glasvezelkabels
- Foutieve uitlijning: glasvezel media zijn niet precies afgestemd op elkaar
- Eind kloof: media raken elkaar niet volledig bij verbinding of aansluiting
- Eind afwerking: media uiteinden zijn niet goed gepolijst of er is vuil aanwezig op beëindiging

Glasvezel vs koper
- Glasvezel is beter maar duurder, moeilijker te installeren en vergt meer veiligheidsmaatregelen.

Draadloos medium zorgpunten
- Dekkingsgebied
- Storingen
- Veiligheid
- Gedeelde media

Soorten draadloze medium
- WI-FI: 11mb-7gb/s
- Bluetooth: 3mb/s, device pairing 1-100 meters
- WI-MAX: 1gb/s, point-to-multipoint

Draadloze LAN benodigheden
- WAP: wireless access point
- Wireles NIC (network interface card) adapter

Datalink
- LLC sublayer
- MAC sublayer
- Verantwoordelijk voor het controleren van de transfer van een frame over de media (toegang verschaffen tot medium)
- Standaarden: IEEE, ANSI, ITU, ISO

WAN topologieën
- Point-to-point: limited to two nodes (source node en destination node), intermediate physical connections zijn mogelijk
- Hub and spoke: één centraal punt (hub)
- Full mesh: alles verbonden met elkaar
- Half-duplex: server -> switch (send)
- Full-duplex: server -> switch (send), switch -> server (receive)

Fysische LAN topologieën
- Star: alles verbonden via één punt
- Extended star: 2 of meer stars verbonden met elkaar
- Bus: één buis waaruit verbindingen gaan
- Ring: cirkel waaruit verbindingen gaan

Contentie gebaseerde toegang
- Versturen als frame volledig klaar is, volledig frame receiven
- CSMA/CD: collision detection, wachten
- CSMA/CA: collision avoidance

Frame
- Structuur van frame en velden in header en trailer variëren naargelang type protocol
- Hoeveelheid controle info toegevoegd aan IP-Packet hangt af van omgeving
- Start en stop frame: indicatie vlaggen begin en einde frame
- Start: frame start, adressing (bron en bestemming-mac), type (indentificeert laag 3 protocol in dataveld), control (indentificeert speciale flow control zoals QoS)
- Data: bevate de frame payload
- Stop: error detectie (fout detectie -> CRC waarde in FCS), frame stop
- LAN-frame: gebruikt lagere bandbreedte technolgie
- WAN-frame: gebruikt hogere bandbreedte technologie

Fysieke adressen
- Uniek per toestel
- Alleen gebruikt voor lokale aflevering

Ethernet
- Eén van de meest gebruikte LAN-tecnhologieën
- Werkt in datalink laag en fysieke laag
- Ondersteunt data bandbreedte van 10, 100, 100, 10.000, 40.000 en 100.00 Mbps

Ethernet standaarden
- Definiëren laag 2 protocollen en laag 1 technologieën
- Twee afzonderlijke sublagen van de datalinklaag om het te laten werken: LLC (logical link control) sublayer, en MAC sublayer

LLC sublayer
- Verzorgt communicatie tussen netwerksoftware en apparaat hardware
- Geïmplementeerd in de software, uitvoering is onafhankelijk van hardware
- Neemt netwerkprotcolgegevens en voegt controle-informatie toe om het pakket naar de bestemming te helpen

MAC sublayer
- Onderste sublaag van datalink-laag
- Uitgevoerd door hardware (meestal in computer NIC)
- IEEE 802.3 normen
- Data-inkapseling
- Media toegangscontrole

MAC data-inkapseling proces
- Samenstellen van de frame voor verzending en ontleden bij ontvangst
- Bij samenstellen voeg MAC-laag een header en een trailer toe aan de netwerklaag PDU
- Frame begrenzing: biedt scheidingstekens om bits van eenzelfde frame the identificieren en biedt synchronisatie tussen zendende en ontvangende knooppunten
- Adressering: elk ethernet header toegevoegd in frame bevat fysiek MAC-adres -> mogelijk frame af te leveren aan zijn bestemmingsknooppunt
- Foutdetectie: checksum cyclische redundantiecontrole (CRC)

MAC media toegangscontrole
- Plaatsen en verwijderen frames op/van medium
- Communiceert rechtstreeks met fysieke laag
- Als meerdere apparaten op een enkel gedeeld medium poging doen gegevens tegelijkertijd te versturen -> botsingen -> beschadigde onbruikbare data
- Ethernet biedt CSMA technologie om toegangsknooppunten te delen
- Bij half-duplex: CSMA/CD detecteert collisions en lost ze op
- Full-duplex switchen: verzenden en ontvangen zonder collisions

Ethernet II velden
- Minimum grootte ethernet frame: 64 bytes (kleiner = runt, beschouwd als collision)
- Maximum grootte ethernet frame: 1518 bytes (excl. preamble) (groter = jumbo of baby giant frame)
- Preamble + SFD (start frame delimiter): gebruikt voor synchronisatie tussen zendend en ontvangend apparaat
- Destination MAC address
- Source MAC address
- EtherType: indentificeert bovenliggend protocol ingekapseld in Ethernet frame (0x800 voor IPv4, 0x806DD voor IPv6 en 0x806 voor ARP)
- Data
- FCS (frame check sequence): fouten detecteren in frame a.h.v. CRC

MAC-adres
- MAC = Media Access Control
- 48 bits, 12 hexadecimale cijfers
- IEEE vereist van verkoper: gebruik maakt van toegewezen OUI als eerste 3 bytes, MAC-adressen metzelfde OUI unieke waarde in 3 laatste bytes
- MAC-adres = BIA (burned-In Address) -> omdat het gebrand wordt in ROM op NIC
- Bij modernere PC besturingssystemen en NICs -> mogelijk MAC-adres te wijzigen
- Bij opstart PC -> MAC-adres van ROM naar RAM
- NIC ontvangt frame en kijkt of bestemmingsMAC-adres overeenkomt met fysiek adres (dan doorgeven of niet)
- Om te verzenden: source en destination IP en MAC-adres nodig
- Unicast: naar één bestemming van switch
- Broadcast: naar alle bestemmingen van switch
- Multicast: naar enkele bestemmingen van switch

Switch
- Laag 2 apparaat
- Neemt doorzend beslissingen op basis van laag 2 ethernet MAC-adressen
- Leeg MAC-adressentabel -> switch stuur bericht naar alle poorten (met uizondering poort waar het vandaan komt)

MAC-adressentabel
- MAC-adressentabel = CAM table = Content Adressable Memory table
- Bericht ontvangen -> kijken naar bron-MAC-adres -> indien nieuw toevoegen aan tabel -> kijken bestemmings-MAC-adres -> kijken in tabel naar welke poort hij moet doorsturen -> indien niet in tabel naar alle poorten sturen behalve waar bericht van kwam (flooding) -> bestemming waarvan MAC-adres overeenkomt met bericht destination-MAC-adres stuurt bevestiging terug -> tabel aangevuld

Doorstuurtechnieken switch
- Store and forward: bij ontvangen frame -> opslaan in buffer tot frame volledig ontvangen is -> tijdens stockage analyseren info bestemming + CRC -> doorsturen naar bestemming (bij QoS nodig!)
- Cut-through: switch heeft bestemmingsadres kunnen lezen -> doorsturen bericht (ook al is het nog niet volledig aangekomen)

Cut-through varianten
- Fast-forward switching: paket onmidellijk na lezen bestemmings-MAC-ares oor gestuurd (ook frames met fouten)
- Fragement-free switching: eerste 64 bytes frame opgeslagen -> fount controle uitvoeren -> doorsturen

Geheugenbuffer switch
- Frames opslaan alvorensd door te sturen, ook als bestemmingspoort bezet is door congestion
- Port-based Memory buffering: frames opgeslagen in queues die gelinkt zijn aan specifieke in en uitgaande poorten
- Shared memory: dumpt alle frames in één memory buffer die alle poorten van de switch delen

Duplex
- Full-duplex: sturen en verzenden tegelijk
- Half-duplex: slechts één uiteinde van de verbinding kan tegelijk verzenden
- Autonegation: optionele functie op ethernet switches/NICs -> Automatisch kiezen voor meest performante
- Duplex mismatch: ene poort half-duplex, andere full-duplex -> performantie probleem door collisions

Auto-MDIX
- mdix auto interface configuration commando -> switch detecteert automatisch kabel-type poort waar hij mee verbonden is en configureert dan de interface met respectievelijke instellingen

Address Resoution Protocol (ARP)
- Twee adressen toegekend aan apparaat in Ethernet LAN: MAC-adres (laag 2, fysiek), IP-adres (laag 3, logisch)
- Bestemmings-IP-adres behoort tot extern netwerk -> bestemmings-MAC-adres -> adres default gateway
- IP-adres geassocieerd aan MAC-adres door ARP
- IPv4-adressen koppelen aan de respectievelijke MAC-adressen
- Onderhouden van tabel met koppelingen

ARP tabel of ARP cache
- Opgeslagen in ram van apparaat
- MAC-adres gekoppeld aan bestemmings-IPv4-adres vinden
- Bestemmingadres in zelfde netwerk als bronadres -> apparaat zoekt in ARP-tabel naar bestemmings-IPv4-adres -> gekoppeld MAC adres opgehaald
- Bestemmingadres in andere netwerk dan bronadres -> apparaat zoekt naar IPv4-adres van default gateway -> gekoppeld MAC adres opgehaald
- Koppeling (of rij) in ARP-tabel = map
- Indien geen map -> ARP aanvraag naar alle -> correct one replies with MAC address
- ARP cache timer: verwijdert maps die al voor een bepaalde tijd niet gebruikt zijn (kan ook handmatig)

ARP veiligheid
- ARP broadcasts met een groot aantal apparaten kan zorgen voor een vermindering in performantie voor een korte tijd
- ARP spoofing/poisoning: ARP-answer met eigen MAC-adres -> hij krijgt packages bedoelt voor iemand anders!

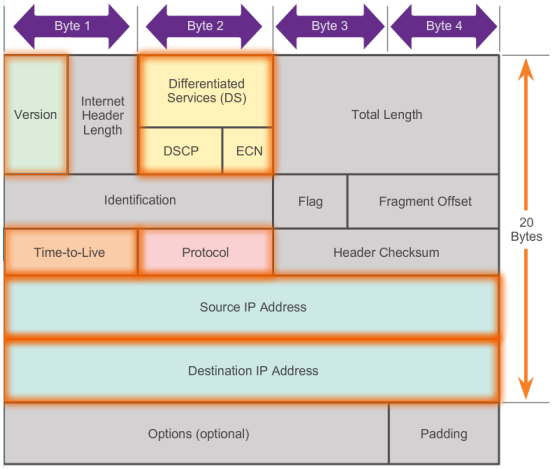
Netwerklaagprotocollen

- Bepalen pakketstructuur en verwerking zodat gegevens andere host kunnen bereiken
- Bestaat uit 4 basisprocessen:
- Adresseren van eindapparaten -> IP-adres voor identificatie in netwerk. Eindapparaat met geconfigureerd IP-adres = host
- Inkapseling -> IP-header info toevoegen (IP-adres van bron en bestemings- host) -> header aan PDU = IP-pakket
- Routering -> paden selecteren die pakket naar bestemmingshost leidt (kan over verschillende tussenschakelapparaten gaan)
- De-inkapselen -> bestemingsIP-adres = eigen IP-adres -> IP-header verwijdert uit pakket
- Gemeenschappelijk: IPv4, IPv6
- Nalatenschap: IPX, AppleTalk, CLNS/DECNet

IP-protocol

- Sender doesn't know if: receiver is present, paccket arrived, receiver can read packet
- Receiver doesn't know: when packet is coming
- Some packets may be lost
- IP is medium onafhankelijk -> maar de maximale grootte van de PDU die een medium kan vervoeren is wel gekend = MTU (maximum transmission unit) -> indien PDU groter dan MTU -> fragmentatie = pakket opsplitsen in via tussenschakelstations

IPv4 header velden



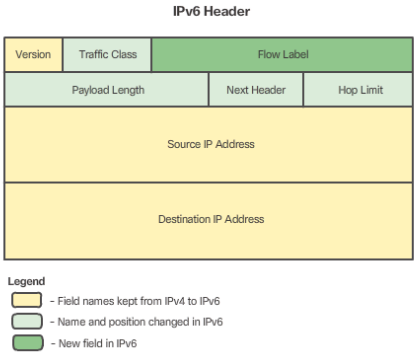
- Version: 4 bits, binaire waarde, indentificatie IP-pakket versie (IPv4 altijd 0100)
- Internet Header Lengt (IHL): 4 bit, binaire waarde, geeft aantal woorden van 32 bits aanwezig in header (20 – 60 bytes)
- Differentiated Services (DS): 8 bit, prioriteit pakket bepalen, 6 bits DSCP waarde voor QoS, 2 bits ECN verlies pakketen voorkomen
- Total length: 16 bit, definieert grootte volledig pakket (incl. header en data in bytes). Min = 20 bytes, max = 65,535 bytes
- Identificiation: 16 bit, unieke identificatie van IP-pakket van orginele pakket
- Flags: 3 bit veld, geeft aan hoe pakket gefragmenteerd is (samen met fragment offsent en identificatie om fragment terug samen te voegen tot oorspronkelijke IP-pakket)

- Fragment Offset: 13 bit, bepaalt volgorde fragmenten
- Time-to-live (TTL): 8 bit, in seconden verwijst meestal naar aantal hops. Verminderd met 1 iedere keer door een hop of router verwerkt. Als TTL = 0-> router verwijdert pakket en stuurt ICMP (internet control message protocol) -> via traceroute commande gebruikte routers tussen bron en bestemming identificeren
- Protocol: 8 bit, type data payload pakket -> netwerklaag gegevens doorgeven aan bovenliggende laag (ICMP, TCP, UDP)
- Header checksum: 16 bit, foutcontrole IP-header
- Source IP Addres: 32 bits, bron-IP-adres van pakket
- Destination IP address: 32 bits, doel-IP-adres van pakket

Beperkingen IPv4
- IP-adres uitputting
- Expansie internet routeringstabel
- Gebrek aan end-to-end connectiviteit door gebruik NAT om tegemoet te komen aan uitputting IP-adressen

IPv6 vs IPv4
- Verhoogde adresruimte: bij IPv6 bestaat IP-adres uit 128 bits (32 hexadecimale cijfers -> aantal mogelijke adressen = 3,4x10^38)
- Betere routing: vereenvoudiging van header en minder velden
- Grotere payload die throughput en transport efficiëntie verhoogt
- Overbodig maken van NAT (network adress translations) tussen private en publieke adressen

IPv6 header velden



- Version: 4 bit, 0110
- Traffic class: 8 bit, 6 bit DSCP, 2 bit ECN
- Flow label: 20 bit, speciale service voor real-time applications, geeft info aan routers en switchen dat ze hetzelfde pad moeten gebruiken voor de pakketstroom zodat er geen herschiking moet gebeuren
- Payload length: 16 bit, grotte van volledige IP-pakket incl. optionele uitbreidingen
- Hop limit: 8 bit, waarde verlagen met 1 voor elke router die pakket doorstuurt -> if = 0 -> pakket weggoiten -> ICMPv6 bericht naar source
- Source address: 128 bit, IPv6-adres verzender
- Destination host: 128 bit, IPv6-adres ontvanger

Default gateway en host pakket forwarding beslissing
- Router verbonden met lokale netwerk segment = default gateway
- Default gateway: routers traffic to other networks, has a local IP address in the same address range as other hosts on the network
- Can take data in and forward data out

IPv4 host routeringstabel
- Commando: netstat –r of route print -> visualiseert routeringstabel van host
- Toont alle gekende IPv4 routes incl.: directe verbindingen, lokaal netwerk, lokaal default routes
- Direct verbonden
- Afstandspaden
- Standaardpaden
- Hoe route werd aangeleerd
- Betrouwbaarheid route
- Waardering route
- Wanneer route werd bijgewerkt
- Welk interface gebruit moet worden om bestemming te bereiken

Lokale route
- Hoe route aangeleerd: C of L (c = direct verbonden, l = lokaal interface)
- Beschrijft bestemmingsnetwerk en hoe verbinden (vb: 192.168.10.0/24 is directly connected)
- Beschrijft interface op router verbonden met bestemmingsnetwerk (vb: GigabitEthernet0/0)

Externe route
- Hoe route aangeleerd
- Adres externe netwerk
- Betrouwbaarheid route bron (vb: [90/)
- Metric (=waarde om externe netwerk te bereiken) (vb: 2710112])
- IP-adres van volgende knooppunt (kan niet doorgestuurd worden als deze niet beschreven staan in routeringstabel)
- Tijd sinds netwerk werd ontdekt
- Uitgaande interface

Cisco routers
- Branch: telewerkers, kleinebedrijven, middelgrote ondernemingen
- WAN: grote bedrijven en organisaties
- Service provider: grote diensverleners

Router geheugen

Geheugen	Vluchtig/Niet-vluchtig	Bevat
RAM/SDRAM	Vluchtig	- Actieve IOS - Actief configuratie bestand - IP-routerings- en ARP-tabellen - Pakket buffer
ROM	Niet-vluchtig	- Bootup instructies - Basis diagnose software (POST) - Beperkte IOS (backup)
NVRAM	Niet-vluchtig	- Startup configuratie bestand
Flash	Niet-vluchtig	- IOS - Andere systeembestanden (log, stemconfig, html, backupconfig –bestanden)

Binnenin router

1 = Voeding
2 = Bescherming WIC (WAN interface card)
3 = Koelvin
4 = SDRAM
5 = NVRAM
6 = CPU
7 = Advanced Integration Module (AIM)

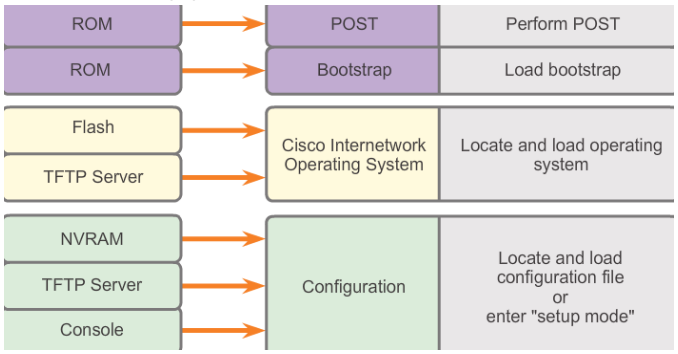
Achterkant router

1 1 2 3 4

5 5 6 7 4 8

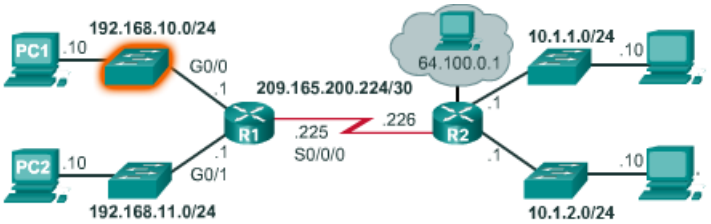
1 = Double-Wide EHWIC slots -> enhanced high-speed WAN interface card
2 = EHWIC 0 (serial WAN interface)
3 = AUX port (management poort)
4 = Lan interfaces (Ethernet)
5 = 2x 4 GB flash card slot
6 = Console USB type B (management poort)
7 = Console RJ45 (management poort)
8 = USB port

Router Bootup proces



Basisconfiguratie switch

- Configure the device name
  - hostname *name*
- Secure user EXEC mode
  - line console 0
  - password *password*
  - login
- Secure remote Telnet / SSH access
  - line vty 0 15
  - password *password*
  - login
- Secure privileged EXEC mode
  - enable secret *password*
- Secure all passwords in the config file
  - service password-encryption
- Provide legal notification
  - banner motd *delimiter message delimiter*
- Configure the management SVI
  - interface vlan 1
  - ip address *ip-address subnet-mask*
  - no shutdown
- Save the configuration
  - copy running-config startup-config




```
Switch>enable
Switch# configure terminal
Switch(config)# hostname S1
S1(config)# enable secret class
S1(config)# line console 0
S1(config-line)# password cisco
S1(config-line)# login
S1(config-line)# line vty 0 15
S1(config-line)# password cisco
S1(config-line)# login
S1(config-line)# exit
S1(config)# service password-encryption
```

Basisconfiguratie router

- Configure the device name
  - hostname *name*
- Secure user EXEC mode
  - line console 0
  - password *password*
  - login
- Secure remote Telnet / SSH access
  - line vty 0 15
  - password *password*
  - login
- Secure privileged EXEC mode
  - enable secret *password*
- Secure all passwords in the config file
  - service password-encryption
- Provide legal notification
  - banner motd *delimiter message delimiter*
- Save the configuration
  - copy running-config startup-config

```
Router>enable
Router#configure terminal
Enter configuration
commands, one per line.
End with CNTL/Z.
Router(config)#hostname R1
R1(config)#
```

OR

```
Router>en
Router#conf t
Enter configuration
commands, one per line.
End with CNTL/Z.
Router(config)#ho R2
R2(config)#
```

```
R1(config)#enable secret class
R1(config)#
R1(config)#line console 0
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)#exit
R1(config)#
R1(config)#line vty 0 4
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)#exit
R1(config)#
R1(config)#service password-encryption
R1(config)#
```

```
R1(config)#banner motd #
Enter TEXT message.  End with the character '#'.

**********************************************
      WARNING: Unauthorized access is
prohibited!

**********************************************
#

R1(config)#
```

```
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

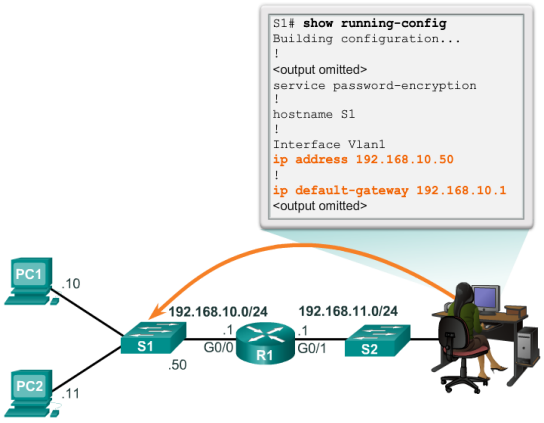
Configuratie van interfaces router

- Configure the interface
  - interface *type-and-number*
  - description *description-text*
  - ip address *ipv4-address subnet-mask*
  - no shutdown

```
R1#conf t
Enter configuration commands, one per line.
End with CNTL/Z.
R1(config)#
R1(config)#interface gigabitethernet 0/0
R1(config-if)#ip address 192.168.10.1 255.255.255.0
R1(config-if)#description Link to LAN-10
R1(config-if)#no shutdown
%LINK-5-CHANGED: Interface GigabitEthernet0/0,
changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0,changed state to up
```

```
R1(config-if)#exit
R1(config)#
R1(config)#int g0/1
R1(config-if)#ip add 192.168.11.1 255.255.255.0
R1(config-if)#des Link to LAN-11
R1(config-if)#no shut
%LINK-5-CHANGED: Interface GigabitEthernet0/1,
changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/1, changed state to up
R1(config-if)#exit
R1(config)#
```

Configureren default gateway op switch



If the default gateway was not configured on S1, response packets from S1 would not be able to reach the administrator at 192.168.11.10. The administrator would not be able to manage the device remotely.

IPv4 voorstellen

- 32 bits
- Uniek
- Binair (11000000 10101000 00001010 00001010) of decimaal (192.168.10.10)

Binair naar decimaal

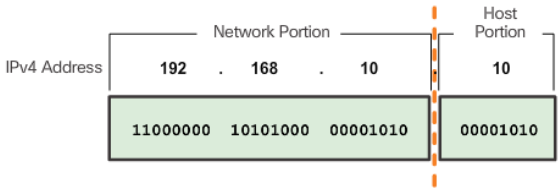
How to convert binary to decimal

The decimal number is equal to the sum of powers of 2 of the binary number's '1' digits place:

binary number:	1		1	1	0	0	1
power of 2:	2^5		2^4	2^3	2^2	2^1	2^0

Vb: $111001_2 = 1{\cdot}2^5+1{\cdot}2^4+1{\cdot}2^3+0{\cdot}2^2+0{\cdot}2^1+1{\cdot}2^0 = 57$

Netwerk- en hostgedeelte

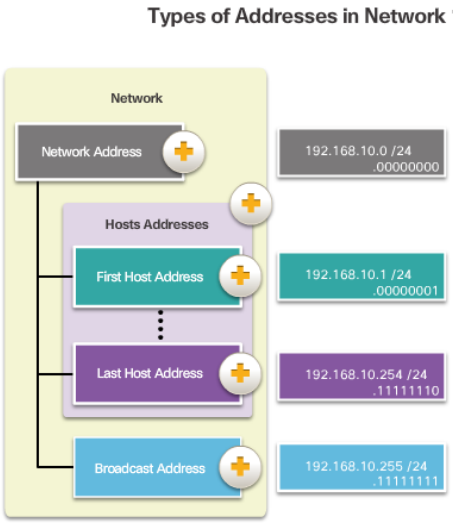


Subnetmask

- Gebruikt om scheiding, ofwel subnet, aan te brengen in de IP-adressering
- Door logische AND operatie tussen IP-adres en subnetmask -> netwerkadres bepaald (en dus netwerkgedeelte)
- Verkorte notatie voor subnet mask:

Subnet Mask	32-bit Address	Prefix Length
255.0.0.0	11111111.00000000.00000000.00000000	/8
255.255.0.0	11111111.11111111.00000000.00000000	/16
255.255.255.0	11111111.11111111.11111111.00000000	/24
255.255.255.128	11111111.11111111.11111111.10000000	/25
255.255.255.192	11111111.11111111.11111111.11000000	/26
255.255.255.224	11111111.11111111.11111111.11100000	/27
255.255.255.240	11111111.11111111.11111111.11110000	/28
255.255.255.248	11111111.11111111.11111111.11111000	/29
255.255.255.252	11111111.11111111.11111111.11111100	/30

Netwerk-, Host-, en Broadcastadres



Toewijen van IPv4-adres aan host

- Statisch: sommige apparaten vereisen een statisch IP-adres (printers, servers, etc.) -> handmatig toewijzen, dit kan ook voor een host
- Dynamisch: automatische toewijzing van IP-adres -> host is DHCP-client en vraag een IP-adres aan DHCP-server ->DHCP-server geeft IP-adres, subnetmask, default gateway en andere configuratie-info

Unicast transmissie

- Host to host communicatie
- Unicast-adres toegepast op eindapparaat = hostadres
- Bronadres van pakket is altijd unicast-adres afkomstig van verzendende host

Broadcast transmissie

- Gebruikt bij veel netwerkprotocollen
- Directed: naar alle hosts vaan een specifiek netwerk
- Limited: verzonden naar 255.255.255.255
- Standaard forwarden routers geen broadcasts

Multicast transmissie
- Host stuurt pakket naar geselecteerde groep hosts die zich abonneren op een multicast-groep
- 224.0.0.0 tot 239.255.255.255 bereik van adressen zijn gereseveeerd voor multicasts

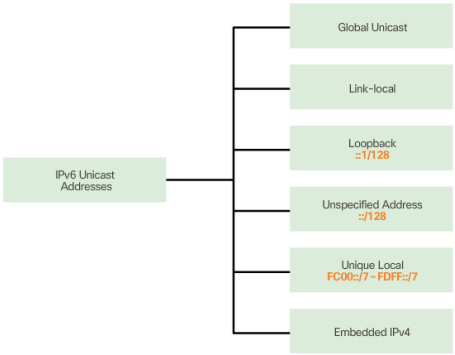
Soorten IPv4 adressen
- Publieke: worden wereldwijd doorgestuurd tussen de ISP (internet service provider) routers.
- Private: blokken van adressen gebruikt door de meeste organisaties om IPv4-adressen toe te wijzen aan interne hosts (NAT wordt gebruikt om Private te vertalen naar Publiek voor internet-toegang)
- Loopback: 127.0.0.0/8
- Link-local (Automatic Private IP Adressing = APIPA): 169.254.0.0/16
- TEST-NET: 192.0.2.0/24

Toekenning IP-adressen
Beheerd door IANA (internet assigned numbers authority) -> verdeelt blokken van IP-adressen aan RIR (regional internet registries) -> toewijzen IP-adressen aan ISP's

Hoe gelijktijdig bestaan van IPv4 en IPv6
- Dual Stack: zorgt dat ze naast elkaar op hetzelfde netwerk blijven bestaan, dual stack apparaten kunnen werken met IPv4 en IPv6-protocol stacks
- Tunneling: methode om IPv6 pakket te transporteren over IPv4-netwerk door IPv6 pakket in te kapsel in een IPv4 pakket
- Translation: NAT64 -> IPv6-apparaten kunnne communiceren met IPv4 –apparaten (omzetten IPv4 naar IPv6 en omgekeerd)

IPv6 adres voorstelling
- 128 bits als reeks van hexadecimale waarden
- Notatie regel: weglaten leidende 0'en

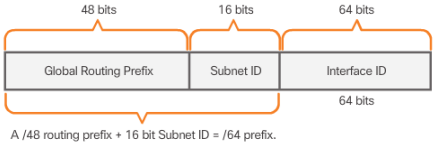
IPv6 Prefix length
- Gebruikt geen dotted-decimal subnetmask notatie
- Duid netwerkgedeelte aan
- Van 0 tot 128 (typisch 64)

IPv6-unicastadres


IPv6 Link-Local Unicastadres

- Apparaat kan communiceren met andere IPv6-apparaaten op dezelfde link (en alleen die link) subnet

IPv6-global-unicastadres



Configuratie global-unicastadres

- Statisch
- Dynamisch via SLAAC, DHCPv6 if via EUI-64

IPv6 Multicast

- Prefix: FF00::/8
- Toegewezen (assigned) of aangevraagd (solicited node)

ICMPv4 en ICMPv6

- Host confirmation
- Destination or service unreachable
- Time exceeded
- Route redirection
- IP is not a reliable protocol -> TCP/IP suite provides message in event of error using ICMP

Pinging

- Local host: TCP/IP is installed and working (127.0.0.1)
- Local LAN: IPv4 connectivity to local network (10.0.0.254)
- AfstandsLAN: connectivity to remote host

Traceroute

- Testing the path

Broadcastdomeinen

- Elke router verbindt een broadcastdomein en broadcasts worden alleen gepropageerd hierbinnen
- Problemen: trage netweroperaties (door hoeveelheid broadcast verkeer) en apparaatoperaties (apparaat moet elk broadcastpakket accepteren en verwerken)
- Oplossing -> verklein grote netwerk door kleinere broadcast domeinen te maken -> subnetting

Redenen voor subnetting

- Groeperen door organisatorische eenheid (vb: administratie, hr, studenten, acccounting)
- Groeperen door apparaat-type (vb: printers, all hosts, all servers)

Subnetting on octet boundary

Subnetting Networks on the Octet Boundary

Prefix Length	Subnet Mask	Subnet Mask in Binary (n = network, h = host)	# of hosts
/8	255.0.0.0	nnnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh 11111111.00000000.00000000.00000000	16,777,214
/16	255.255.0.0	nnnnnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh 11111111.11111111.00000000.00000000	65,534
/24	255.255.255.0	nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.00000000	254

Voorbeelden:

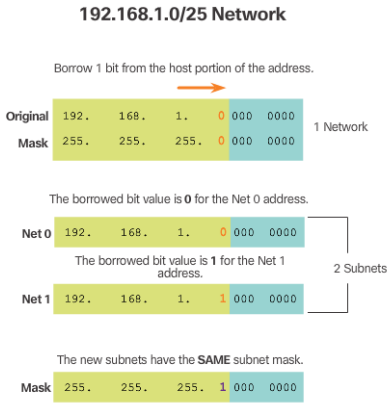
Subnet Address (256 Possible Subnets)	Host Range (65,534 possible hosts per subnet)	Broadcast
10.0.0.0/16	10.0.0.1 - 10.0.255.254	10.0.255.255
10.2.0.0/16	10.2.0.1 - 10.2.255.254	10.2.255.255
10.3.0.0/16	10.3.0.1 - 10.3.255.254	10.3.255.255
10.4.0.0/16	10.4.0.1 - 10.4.255.254	10.4.255.255
10.5.0.0/16	10.5.0.1 - 10.5.255.254	10.5.255.255
10.6.0.0/16	10.6.0.1 - 10.6.255.254	10.6.255.255
10.7.0.0/16	10.7.0.1 - 10.7.255.254	10.7.255.255
...	...	...
10.255.0.0/16	10.255.0.1 - 10.255.255.254	10.255.255.255

Subnet Address (65,536 Possible Subnets)	Host Range (254 possible hosts per subnet)	Broadcast
10.0.0.0/24	10.0.0.1 - 10.0.0.254	10.0.0.255
10.0.1.0/24	10.0.1.1 - 10.0.1.254	10.0.1.255
10.0.2.0/24	10.0.2.1 - 10.0.2.254	10.0.1.255
...	...	...
10.0.255.0/24	10.0.255.1 - 10.0.255.254	10.0.255.255
10.1.0.0/24	10.1.0.1 - 10.1.0.254	10.1.0.255
10.1.1.0/24	10.1.1.1 - 10.1.1.254	1.1.1.0.255
10.1.2.0/24	10.1.2.1 - 10.1.2.254	10.1.2.0.255
...	...	...
10.100.0.0/24	10.100.0.1 - 10.100.0.254	10.100.0.255
...	...	...
10.255.255.0/24	10.255.255.1 - 10.255.255.254	10.255.255.255

Classles subnetting

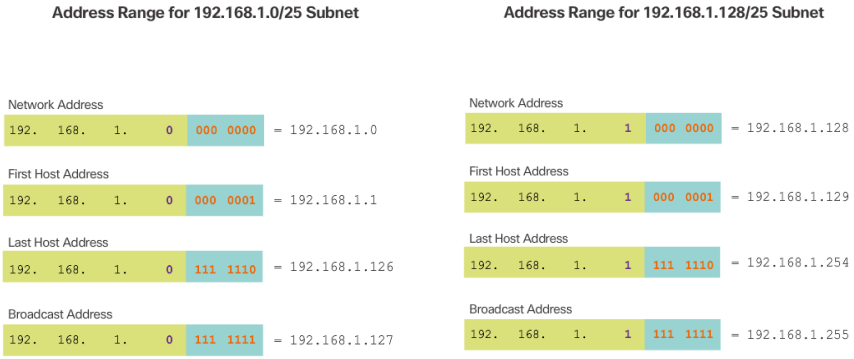
Prefix Length	Subnet Mask	Subnet Mask in Binary (n = network, h = host)	# of subnets	# of hosts
/25	255.255.255.128	nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.10000000	2	126
/26	255.255.255.192	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnhhhhhh 11111111.11111111.11111111.11000000	4	62
/27	255.255.255.224	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000	8	30
/28	255.255.255.240	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnhhhh 11111111.11111111.11111111.11110000	16	14
/29	255.255.255.248	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnhhh 11111111.11111111.11111111.11111000	32	6
/30	255.255.255.252	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnnhh 11111111.11111111.11111111.11111100	64	2

Voorbeeld:



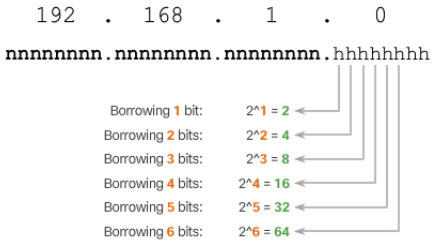
Meerdere subnetten

Voorbeeld:



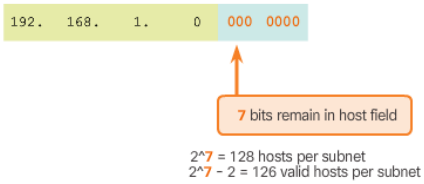
Aanta subnetten berekenen

2^n (n = bits borrowed)

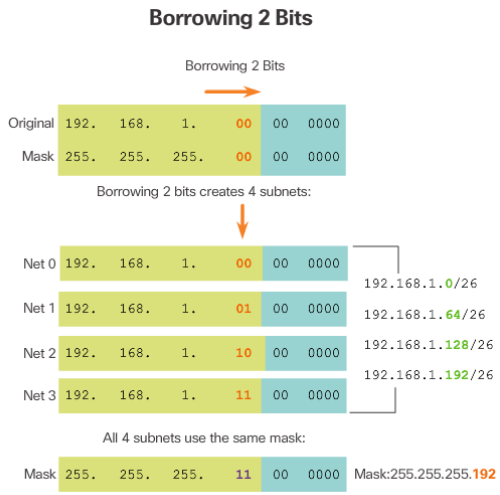


Aantal hosts berekenen

2^n-2 (n=number of bits remaining in the host field)



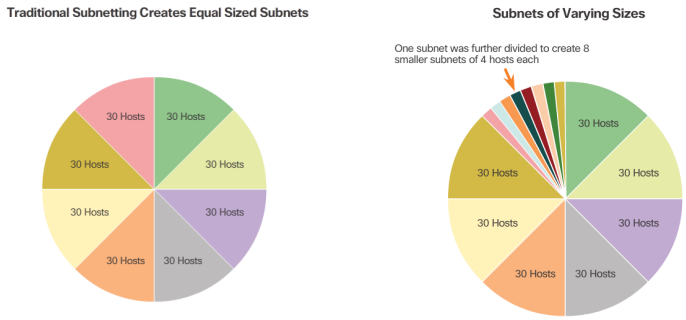
Creëren van 4 subnetten



Subnet gebaseerd op eisen

- Aantal hostadressen nodig
- Aantal subnetten nodig
- Hoe meer bits geleend worden om subnetten te creëren hoe minder host bits beschikbaar

Traditional subnetting vs VLSM



VLSM

Basic Subnetting of 192.168.20.0/24

	/27 Network	Hosts
Building A	.0	.1 - .30
Building B	.32	.33 - .62
Building C	.64	.65 - .94
Building D	.96	.97 - .126
WAN R1 – R2	.128	.129 - .158
WAN R2 – R3	.160	.161 - .190
WAN R3 – R4	.192	.193 - .222
Unused	.224	.225 - .254

VLSM Subnetting of 192.168.20.0/24

	/27 Network	Hosts
Bldg A	.0	.1 - .30
Bldg B	.32	.33 - .62
Bldg C	.64	.65 - .94
Bldg D	.96	.97 - .126
Unused	.128	.129 - .158
Unused	.160	.161 - .190
Unused	.192	.193 - .222
	.224	.225 - .254

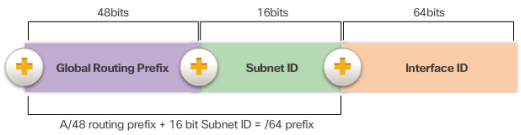
	/30 Network	Hosts
WAN R1–R2	.224	.225 - .226
WAN R2–R3	.228	.229 - .230
WAN R3–R4	.232	.233 - .234
Unused	.236	.237 - .238
Unused	.240	.241 - .242
Unused	.244	.245 - .246
Unused	.248	.249 - .250
Unused	.252	.253 - .254

Waar rekening meehouden

- Preven duplication of addresses
- Monitor security and performance
- Provide control en access

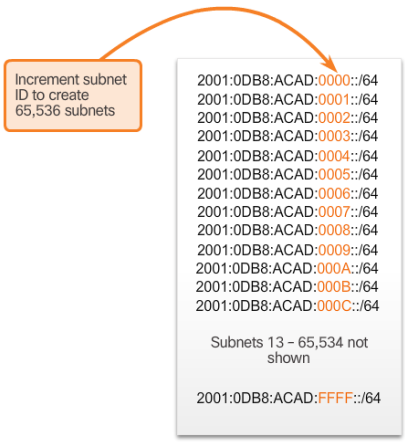
IPv6 Global Unicast adres

IPv6 Global Unicast Address Structure



IPv6 subnetting door gebruik Subnet ID

Address Block: 2001:0DB8:ACAD::/48



Transportlaag protocollen

- Verantwoordelijk voor oprichten tijdelijke communicatie sessie tussen 2 applicaties en het leveren van data hiertussen
- Connection-oriented data stream ondersteuning
- Betrouwbaarheid
- Flow control
- Multiplexing

Verantwoordelijkheden transportlaag

- Individuele gesprekken volgen
- Segmenteren en samenvoegen data
- Applicaties identificeren

Individuele gesprekken volgen

Door apart bijhouden van elk individuele gespreksdatastroom tussen een bron- en bestemmingsapplicatie

Segmenteren en samenvoegen data

- Data opsplitsen in blokken met geschikte grootte -> gemakkelijker beheren en transporteren
- Ieder blok omgevormd naar een segment door toevoeging header om blokken terug samen te voegen

Identificeren van applicaties

- Zorgen dat elke applicatie de juiste data krijgt (ook al draaien er verschillende applicaties tegelijk)

Multiplexen van gesprekken

- Segmenteren van data laat veel verschillende communicaties toe van veel verschillende gebruikers door deze te verweven (=multiplexen) op hetzelfde netwerk
- Transportlaag voegt een header toe met binaire data -> elk segment kunnen identificeren -> verschillende transportprotocollen kunnen hun functies in het beheer van datacommunicatie uitvoeren

Betrouwbaarheid transportlaag

- TCP/IP suite biedt 2 transportlagen: TCP, UDP
- IP gebruikt deze transport-protocollen om het mogelijk te maken voor hosts om met elkaar te communiceren en data uit te wisselen
- TCP (Transmission Control Protocol): betrouwbaar, full featured -> bevestiging verstuurd bij ontvangst data (FTP, http, SMTP, DNS)
- UDP (User Datagram Protocol): geen betrouwbaarhied, heel eenvoudig (DNS, TFTP)

TCP

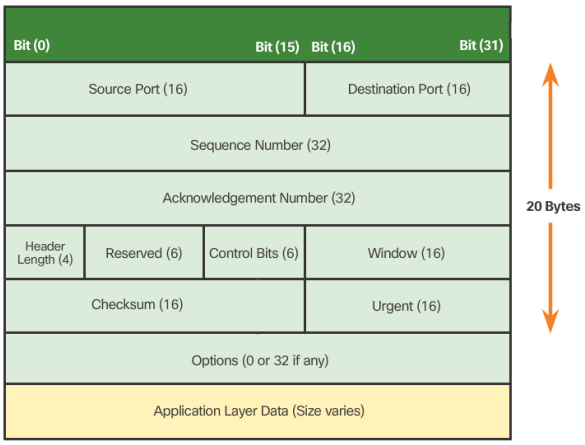
- Nummeren en volgen van segmenten verzonden naar een specifieke host komende van een specifieke toepassing
- Bevestigen van ontvangen data (=acknowlegdement)
- Opnieuw verzenden van elk onbevestigd segment na een bepaalde tijd
- Voor: databases, web browsers, email clients

UDP

- Soms geen betrouwbaarheid nodig
- Sneller dan TCP (minder kans op vertragingen in transmissie door overhead)
- Overhead kan voor sommige toepassingen de bruikbaarheid verminderen of zelfs schadelijk zijn
- Voor: live audio, live video, VoIP

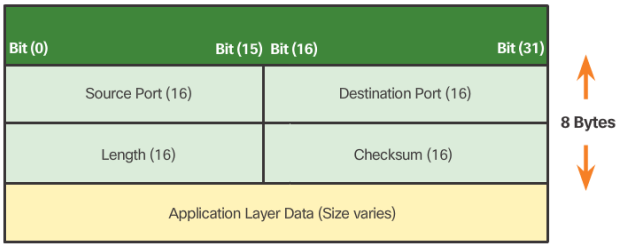
Kenmerken TCP

- Segmentatie + samenvoegen
- Oprichting van een sessie: zekerheid dat applicatie klaar is om data te ontvangen
- Betrouwbare levering: verlorgen segmenten opnieuw zenden
- Aflevering in dezelfde volgorde: segmenten in juiste orde
- Flow control: zorgt dat ontvanger data kan verwerken
- Stateful: houdt toestand bij van communicatie-sessie
- TCP-segment heeft 20 bytes aan overhead in de header die de data van de applicatielaag inkapselt



Kenmerken UDP

- Geen sessie oprichten
- Geen betrouwbare levering, moet worden behandeld door de toepassing
- Geen flow control
- Stateless: houdt toestand niet bij van communicatie-sessie
- Stukken communicatie in UDP = datagrammen
- UDP kleine overhead: 8 byte



Meerdere afzonderlijke gesprekken

- Meerdere communicaties met verschillende transport vereisten scheiden en beheren
- Verschillende toepassignen verzenden en ontvagen data gelijktijdig via netwerk
- Unieke headerwaarden zorgen ervoor dat TCP en UDP deze meervoudige, gelijktijdige gesprekken kunnen beheren door deze toepassingen te identificeren a.d.h.v. poortnummers

Bronpoortnummers

- Dynamisch gekozen door verzendend apparaat om gesprek te identificeren tussen 2 apparaten
- http-client verstuurt meestal meerdere http-veroeken naar een webserver op hetzelfde moment. Elk afzonderlijk http-gesprek wordt bijgehouden op basis van bronpoorten

Bestemmingspoortnummers

- Gebruikt om toepassing of draaiende dienst (service) in server te identificeren
- Server kan meer dan één dienst aanbieden op hetzelfde moment (vb: webservice op poort 80, FTP op poort 21)

Socket

- Combinatie bron-IP-adres en bronpoortnummer of bestemmings-IP-adres en bestemmingspoortnummer -> socket
- Gebruikt om server en dienst aangevraagd door client te identificeren
- Twee sockets worden gecombineerd om een socketpaar te vormen (vb: 192.168.1.5:1099, 192.168.1.7:80)
- Door sockets kunnen meerdere processen die draaien op een cliënt, als meerdere verbindingen met een server proces, zich onderscheiden van anderen.
- Bronpoortnummer fungeert als terugkeeradres voor verzoekende toepassing
- Taak van transportlaag om actieve sockets te blijven volgen

Groepen van poortnummers

- IANA is verantwoordelijk voor toewijzen van verschillende adresseringsnormen o.a. poortnummers

Port Number Range	Port Group
0 to 1023	Well-known Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

Port Number	Protocol	Application	Acronym
20	TCP	File Transfer Protocol (data)	FTP
21	TCP	File Transfer Protocol (control)	FTP
22	TCP	Secure Shell	SSH
23	TCP	Telnet	–
25	TCP	Simple Mail Transfer Protocol	SMTP
53	UDP, TCP	Domain Name Service	DNS
67, 68	UDP	Dynamic Host Configuration Protocol	DHCP
69	UDP	Trivial File Transfer Protocol	TFTP
80	TCP	Hypertext Transfer Protocol	HTTP
110	TCP	Post Office Protocol version 3	POP3
143	TCP	Internet Message Access Protocol	IMAP
161	UDP	Simple Network Management Protocol	SNMP
443	TCP	Hypertext Transfer Protocol Secure	HTTPS

Netstat commando

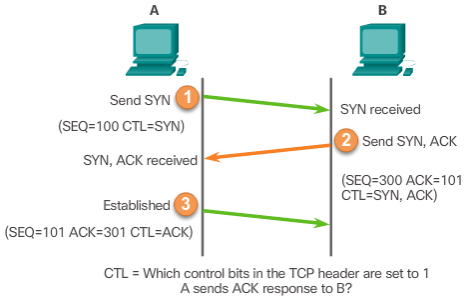
- Onverklaarbare TCP verbindingen -> veiligheidsbedreiging
- Netstat gebruiken om actieve verbindingen in host te controleren
- List van protocollen in gebruik, locale adressen, poortnummers en status van verbinding
- Netstat probeert IP-adressen om te vormen naar domeinnamen en poortnummers naar toepassingen
- -n kan gebruikt worden op IP-adressen/poortnummers in hun numerische vorm te tonen

TCP server processen

- Elk proces dat draait op de server gebruikt een poortnummer
- Individuele server kan geen diensten hebben die toegewezen zijn aan hetzelfde poortnummer binnen dezelfde transportlaagdienst
- Een actieve serverapplicatie toegewezen aan een specifieke poort -> open
- Elk inkomend client verzoek naar open poort -> aanvaard en verwerkt door serverapplicatie verbonden met deze poort
- Er kunnen terzelfdetijd veel poorten open zijn op een server, één voor elke actieve serverapplicatie

Opstellen TCP verbinding

- Initiërende client verzoekt een client-to-server communicatie-sessie met server
- Server stemt toe en vraagt om een server-to-client communciatie-sesse
- Initiërende client bevestigd

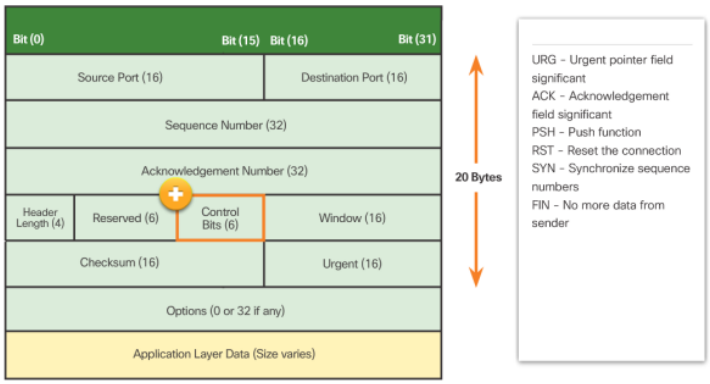


Beëindigen TCP sessie

- FIN TCP vlag met waarde 1 verzonden als de client geen data in de stream te verzenden heeft
- Server zendt een ACK om te bevestigen dat hij de FIN met waarde 1 ontvangen heeft
- Server zendt FIN naar client om de server-to-client sessie te beëindigen
- Client antwoord met ACK om de FIN van de server te bevestigen
- Als alle segmenten bevestigd zijn -> sessie gesloten

TCP Three-Way Handschake

- Stelt vast dat bestemmingsapparaat aanwezig is in netwerk
- Controleert of bestemmingsapparaat ene actieve dienst heeft en de verzoeken aanvaardt op bestemmingspoortnummer die initiërende client wil gebruiken
- Informeert bestemmingsapparaat dat bron cliënt van plan is een communicatiesessie op te stellen op dat poortnummer



TCP betrouwbaarheid – geordende aflevering

- Segmenten gebruiken volgnummers om elk segment eenduidig te identificeren, bevestigen, volgorde van segmenten te volgen en aan te duiden hoe ze terug moeten samengevoegd worden
- ISN (initial sequence numer) wordt willekeurig gekozen bij opstellen TCP sessie -> vervolgens verhoogd met aantal verzonden bytes
- Ontvangend TCP process buffert de segment data tot alles ontvangen is en voegt die dan samen
- Segmenten ontvangen in verkeerde volgorde worden achtergehouden om later te verwerken
- Data wordt doorgegeven aan applicationlaag wanneer alles ontvangen werd en dan terug samengevoegd

TCP betrouwbaarheid – volgnummers en bevestigingen

- Ieder segment dat bestemming bereikt bevestigen
- Verzekert dat bestemming bereikbaar is en klaar om data te ontvangen
- Bestemmingshost bevestigt de ontvangen data
- Opnieuw verzenden van verloren segmenten
- Alle ontvangen segmenten terug in juiste volgorde
- Deftige beëindiging wanneer er geen data meer moet voorzien worden (FIN vlag)
- TCP eindpunt kan ook abrupt sessie beeïndigen indien nodig (RST vlag)

SACK (sekective acknowledgement)

- Optioneel
- Indien beide hosts SACK ondersteunen -> bestemming kan discontinue segmenten bevestigen -> host moet alleen ontbrekende data opnieuw verzenden

TCP Flow Control – Window Size en bevestigingen

- Zorgt dat TCP-eindpunten data op een betrouwbare manier kunnen ontvangen en verwerken
- Door aanpassen snelheid waarmee gegevensstroom tussen bron en bestemming verstuurd wordt voor een bepaalde sessie
- Gebaseerd op een 16-bit-TCP-headerveld nl. Window Size -> is het aantal bytes dat het bestemmings-apparaat in één keer kan aannemen en verwerken
- TCP bron en bestemming bepalen windows size als TCP-sessie tot stand gebracht is
- TCP eindpunten kunnen de window size aanpassen tijdens sessie indien nodig

TCP Flow Control – vermijden van congestie

- Congestie -> weggooien van pakketten
- Niet afgeleverd TCP-segment -> hertransmissie -> kan congestie erger maken
- Bron kan een bepaald niveau congestie inschatten door te kijken naar de snelheid waarmee TCP-segmenten worden verstuurd, maar niet bevestigd
- Bron kan aantal verzendende bytes verminderen vooraleer het een bevestiging op congestie detectie onvangt
- Bron vermindert aantal onbevestigde bytes dat hij verstuurd NIET de windows size
- Bestemming is meestal niet van netwerkcongestie dus zal ook geen nieuwe windows size suggereren

UDP lage overhead vs betrouwbaarheid

- Simpel protocol
- Basis transportlaag functies
- Veel minder overhead dan TCP
- Niet connection-oriented en levert geen ingewikkelde hertransmissie, volgorde bepalding en flow control mechanismen
- Applicaties die UDP gebruiken kunnen betrouwbaarheid gebruiken -> ingebouwd in applicatielaag
- Ontworpen om eenvoudiger en sneller te zijn dan TCP ten koste van betrouwbaarheid
- Voegt data toe in volgorde waaarop ze verzonden waren -> applicatie moet juiste volgorde detecteren indien nodig

UDP server processen

- Well-known of gerigstreerde poortnummers toegewezen aan UDP-gebaseerde serverapplicaties
- UDP-toepassingen en diensten die op server draaien accepteren UDP client verzoeken
- Verzoeken ontvangen op een specifieke poort worden doorgestuurd naar de juiste toepassing op basis van poortnummers

UDP client processen

- UDP client-server communicatie is ook geïnitieerd door een client-toepassing
- Client-proces kiest dynamisch een poortnummer als bronpoort
- Bestemmingspoort is meestal well-known of geregistreerd poort nummer toegewezen aan het serverproces
- Hetzelfde bronbestemmingspaar van poorten worden gebruikt in de header van alle datagrammen in de transactie
- Gegevens die terugkeren naar de client van de server maken gebruik van een verwisseling tussen bron- en bestemmingspoortnummers in de header van het datagram

Applicatielaag protocollen

- Bestaat uit: sessie, presentatie, applicatie
- Situeert zich het dichtst bij de eindgebruiker
- Netwerkapplicaties maken het mogelijk voor gebruikers om gemakkelijk data te verzenden en te ontvangen
- Gedraagt zich als interface tussen applicaties en onderliggend netwerk
- Helpen bij uitwisseling data tussen programma's die draien op bron-en bestemmingshost
- TCP/IP applicatielaag voert functies van bovenste 3 lagen van OSI model
- Vaak gebruikte: http, FTP, TFTP, DNS

Presentatielaag

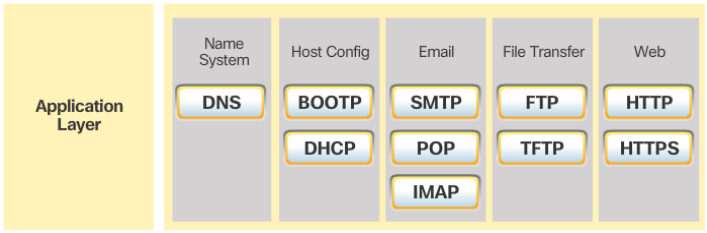
- Data formateren
- Data comprimeren
- Data encrypteren
- Video: QuickTime, Motion Picture Experts Group (MPEG)
- Beeld: Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG), Portable Network Graphics (PNG)

Sessielaag

- Creëert en onderhoudt dialogen tussen bron en bestemmingsapplicaties
- Uitwisseling van info om dialogen te starten, actief te houden en verstoorde of nutteloze sessies opnieuw op te starten

TCP/IP Applicatielaagprotocollen

- Specifiëren het format en de controle info die nodig is voor veel gebruikte internetfuncties
- Moeten zowel op bron- als bestemmingsapparaat geïmplementeerd zijn
- Op bron- en bestemmingshost moeten compatibel zijn om te kunnen communiceren



Client-server model

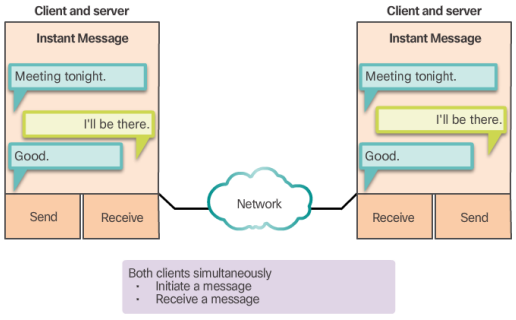
- Apparaat die info aanvraagt = client
- Apparaat die aanvraag beantwoordt = server
- Client- en serverprocessen deel van applicatielaag
- Client initieert uitwisseling door vragen van gegevens server
- Server antwoordt door sturen één of meerdere streams van data naar client
- Applicatielaagprotocollen beschrijven: formaat aanvraag, antwoord tussen client en server
- Inhoud data hangt af van gebruikte applicatie
- Vb: Email (client-server interactie)
- Apparaat download dus iets van de server

Peer-to-peer netwerken

- Data wordt verkregen zonder gebruik te maken van een specifiek toegewezen server
- Twee of meer computers kunnen verbonden zijn met P2P-netwerk om elkaars bronnen te delen
- Elk verbonden eindapparaat (a peer) kan fungeren als client en server
- Rol client en server wordt ingesteld op basis van verzoek

P2P-hybride

- Delen van bronnen gedecentraliseerd
- Indexen die verwijzen naar bronlocaties opgeslagen in centrale directory
- Elke peer heeft toegang tot een index-server om locatie te verkijgen van een bron die zich op een andere peer bevindt.



Veel gebruikte P2P applicaties

- Voorbeelden: eDonkey, G2, BitTorrent, Bitcoin, …
- Mogelijkheid om stukken van verschillende bestanden terzelfdertijd te delen met elkaar
- Klein snel fluctuerend bestand bevat info over locatie van andere gebruikers en opsporingscomputers (trackers)
- Trackers zijn computers die het spoor bijhouden van bestand gehost door gebruikers
- Deze technologie = BitTorrent
- Voorbeelden BitTorrent-clients: BitTorrent, uTorrent, Frostwire, qBittorrent, etc.

HTTP (HyperText Transfer Protocol)

- Webadres of URL (Uniform Resource Locator) = verwijzing naar webserver -> maakt verbinding mogelijk
- URL's en URI's (Uniform Resource Identifier) zijn namen waarmee meeste mensen webadressen associëren
- Voorbeeld delen URL: http = protocol, www.thisisasite.com = de servernaam, index.html = specifieke bestandsnaam
- Bij gebruik DNS -> servernaam vertaalt naar geassocieerd IP-adres -> server contacteren

HTML (HyperText Markup Language)

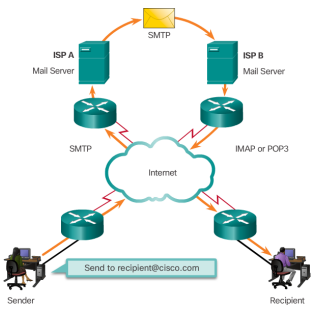
- Browser verstuurt GET verzoek naar server's IP-adres en vraagt bvb: index.html
- Server zendt het verzochte bestand naar de client
- index.html was gespecificieerd in de URL en bevat HTML code voor deze webpagina
- Browser verwerkt HTML-code en formatteert de pagina voor het browservenster op basis van code in bestand

HTTP en HTTPS

- HTTP: verzoek/antwoord protocol, boodschap-types: GET, POST, PUT, niet veilig (berichten kunnen onderschept worden)
- HTTPS: gebruikt authenticatie en encryptie om data te beveiligen

Email protocollen

- Store-and-forward methode van verzenden, opslaan en ophalen van elektronische berichten
- Opgeslagen in databases op mailservers
- Email clients communiceren met mailservers om email te verzenden/op te halen
- Mailservers communiceren met andere mailservers om berichten te transporteren van het ene domein naar het andere
- Email clients communiceren niet direct bij het verzenden van e-mail
- Email is gebaseerd op 3 protocollen voor uitvoering: SMTP (verzenden), POP (ophalen), IMAP (synchroniseren)

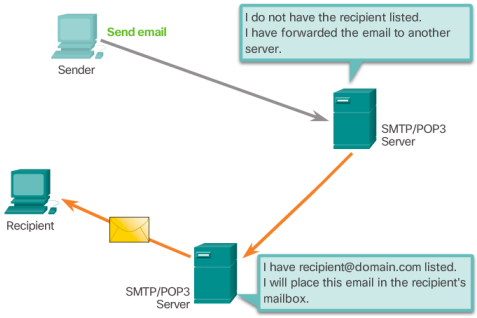


SMTP

- Requires message header and body
- Body can contain any amount of tekst
- Header must have: properly formatted recipient and sender email address
- SMTP client sends email by connecting to SMTP server on port 25
- Server receives message and stores it in a local mailbox or relays the message to another mail server
- Users use email clients to retrieve messages stored on server
- IMAP and POP are 2 protocols commonly used by email clients to retrieve messages

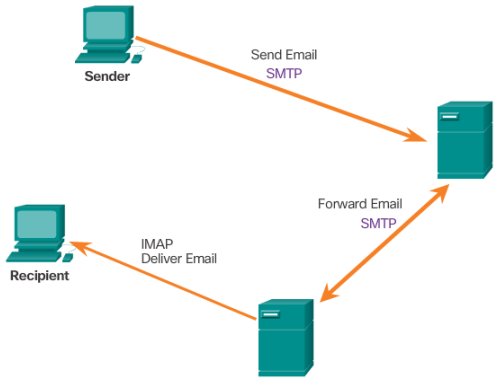
POP

- Messages are downloaded from the server to the client
- Server listens on port 110 TCP for client requests
- Email clients direct their POP requests to mail servers on port TCP 110
- POP client and server exchange commands and responses until the connection is closed or aborted
- POP allows for email messages to be downloaded to the clients device and removed from the server
- There is no coentralized location where email messages are kept
- A downloaded message resides on the device that triggered the download



IMAP operation

- Also used to retrieve email messages
- Allows for messages to be displayed to the user rather than downloaded
- Original message resides on the server until manually deleted by the user
- Users view copies of the messages in their email client software
- Users can create a folder hierarchy on the server to organize and store mail
- That file structure is displayed on the email client
- When user decides to delete a message the server synchronizes that action and deletes the message from the server



Domain Name Service (DNS)

- IP-addresses are crucial for network communication but hard to remember
- Domain names are created to make server addresses more user-friendly
- The domain name is associated with the IP-address of a specific server
- Computer still needs the actualy numeric address before they can communicate
- DNS allows for dynamic translation of a domain name into the correct IP address
- DNS protocol communications use a single format called a message