

Network Engineering Deel 2

Inhoudsopgave

- Routing & Switching
 - Week 4 STP
 - Switching Loops
 - Protocol 802.1 D
 - Voorbeeld
 - BDPU
 - Gebruik
 - Werking
 - Fase 1
 - Fase 2
 - Fase 3
 - Port roles
 - Week 5 Routing intro, MLS, SVI's
 - IP Routing
 - SVI (Switched Virtual Interfaces)
 - Inter VLAN-Routing
 - Voor en Nadelen
 - Routed ports
 - Week 6 VTP (VLAN Trunk Protocol)
 - Voordelen en nadelen
 - Revisienummer
 - Pruning
 - Modes
 - DHCP
 - Werking
 - Configuratie
 - Relay
 - Oplossing: IP Helper Address
 - Redundantie
 - HSRP (Hot Standby Routing Protocol)
 - Progression
 - Operation
 - MAC-adres
 - Configuratie
 - Spanning Tree (HSRP)
 - Week 7 ACLs, NAT

- ACL Acces Control List
 - Type ACLs
 - Configuratie
 - Inbound of Outbound
 - Standard of Extended
 - Standaard
 - Extended
 - De drie Per's
- NAT (Network Address Translation)
 - PAT
 - Configuratie
 - Port forwarding

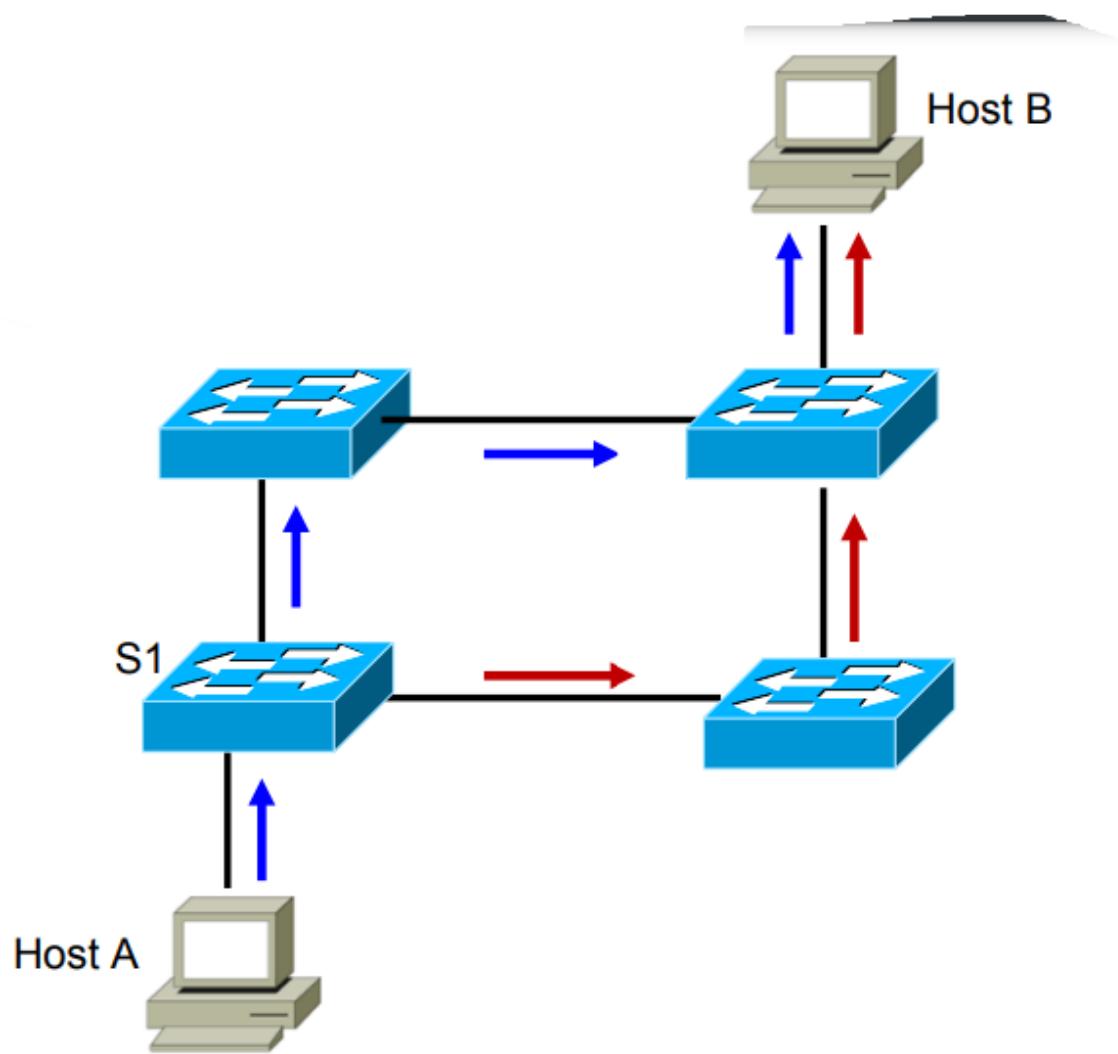
Routing & Switching

Week 4 STP

Switching Loops

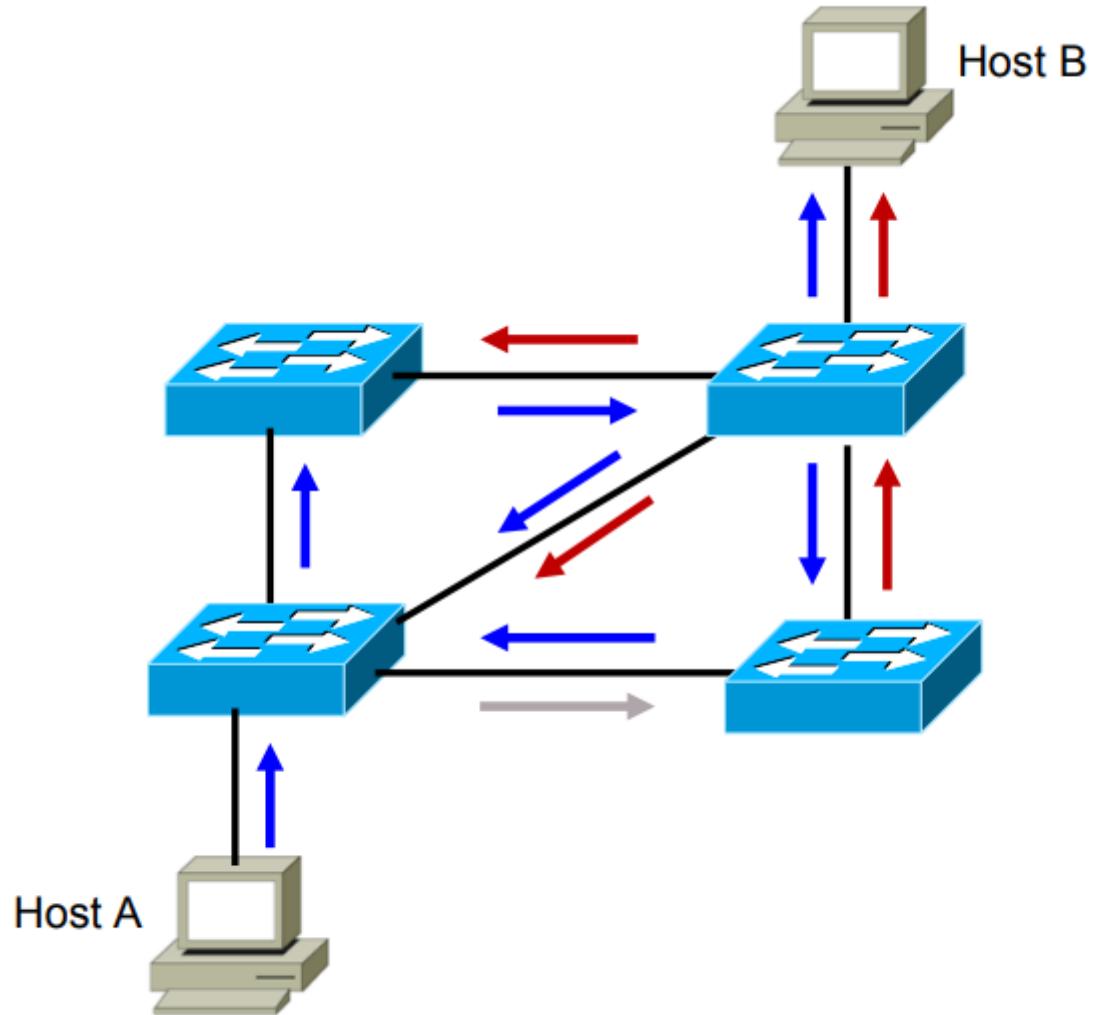
1. Multiple Frames

- Intentieel geen entries in mac-table S 1
- Host A zend een frame naar host B
- S 1 (lege mac-table) flood de switch
- Host B zal nu twee frames ontvangen



2. Broadcast Storm

- Host A zend een broadcast frame
- Dit wordt ge-flood door alle switches
- Een broadcast storm resulteert



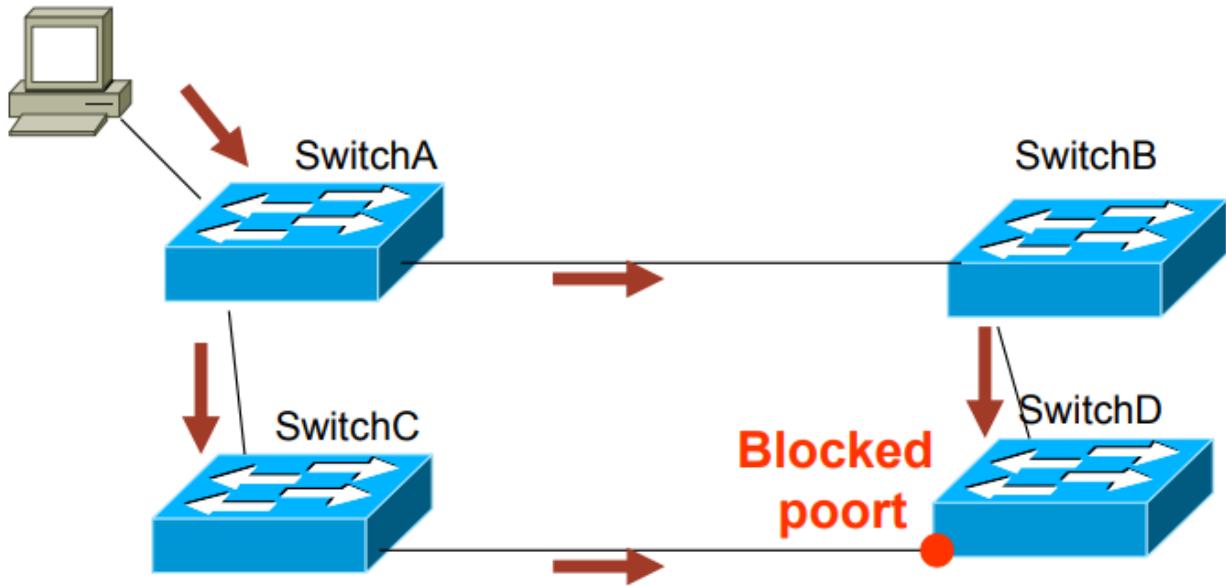
Protocol 802.1 D

Bij **Switching Loops** werden problemen gegeven die kunnen voorkomen bij het aansluiten van meerdere switches. Deze problemen zijn op te lossen met **STP**, met een **Spanning Tree Algoritme** wordt de LAN verdeeld in een **boomstructuur**.

Daarna worden de beste paden gekozen door het netwerk en sommige paden worden afgesloten hierdoor heb je geen loops meer in het netwerk.

Voorbeeld

Door een van de poorten , in het broadcast domain, te blokkeren voorkom je dat er switching loops voorkomen. Dit is omdat een blocked switchport geen user data verstuurt.



BDPU

STP maakt gebruik van **BPDU** (Bridge Protocol Data Units)

Deze STP-frames worden tussen switches gestuurd om de "*Spanning Tree*" op te bouwen. En bepaalt welke paden het best zijn en welke poorten geblokkeerd moeten worden.

ROOT-BID (64) =
ROOT PRIORITY (16) + ROOT MAC(48)

ROOT-PATH-COST (16)

SENDER-BID (64) =
BRIDGE PRIORITY (16) + BRIDGE MAC(48)

PORT-ID (16) =
PORT PRIORITY(8) + PORT NUMBER(8)

Belangrijke informatie voor BPDU is dat je een BID (Bridge Identifier) hebt die bestaat uit de **Priority en MAC-adres**. Ook heb je een priority range van 1 t/m 32768 (in stappen van 4096) met default waarde **32768**

Gebruik

Bij het ontvangen van de BPDU zal in een switch de volgende stappen genomen worden:

#Handigvoorpdetoets

1. De laagste waarde wint

2. Priority waardes worden eerst vergeleken, als gelijk dan wordt naar het MAC-gekeken (de bridge-ID bestaat uit deze twee componenten, ze worden niet opgeteld).
3. BID is belangrijker dan
root-path-cost is belangrijker dan
port-ID

Werking

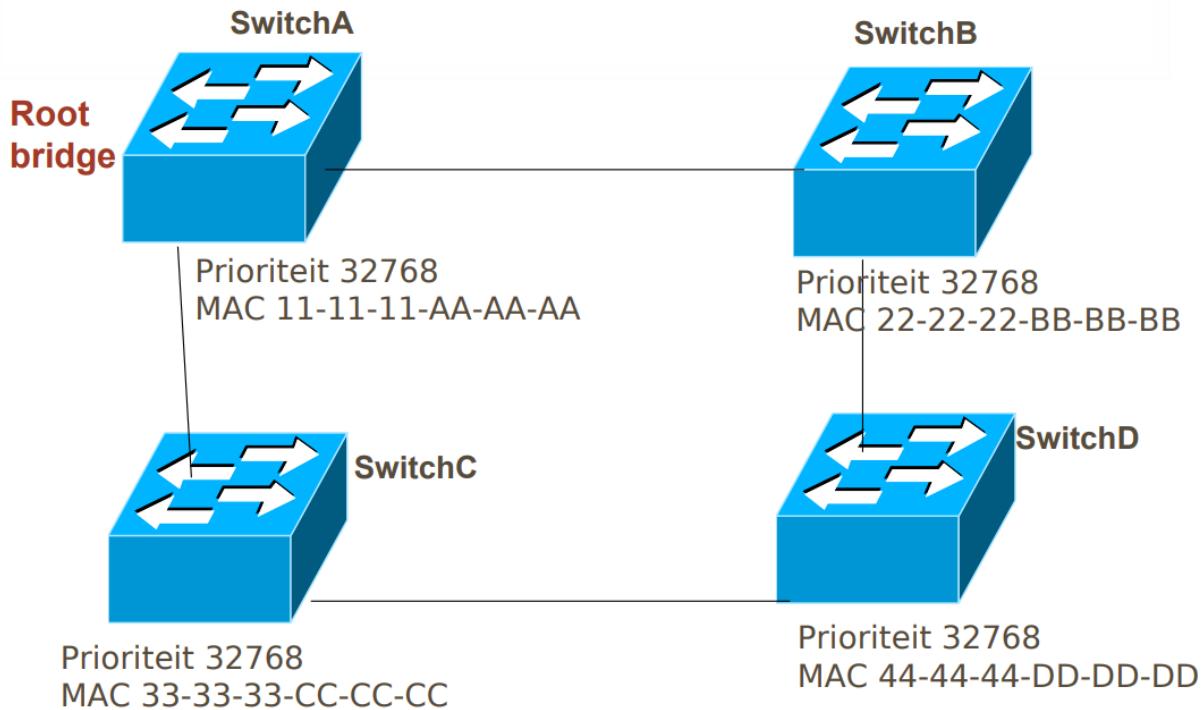
De manier waarop STP werkt kan in 3 fasen worden uitgelegd

1. De **root bridge**(*root war*) wordt gekozen
2. De **root ports**(ports die "wijzen" naar de root bridge) op de non-root bridges
3. De **designated ports**(ports die wijzen vanaf de root bridge) per segment worden gekozen en de overgebleven ports worden **geblokkeerd**

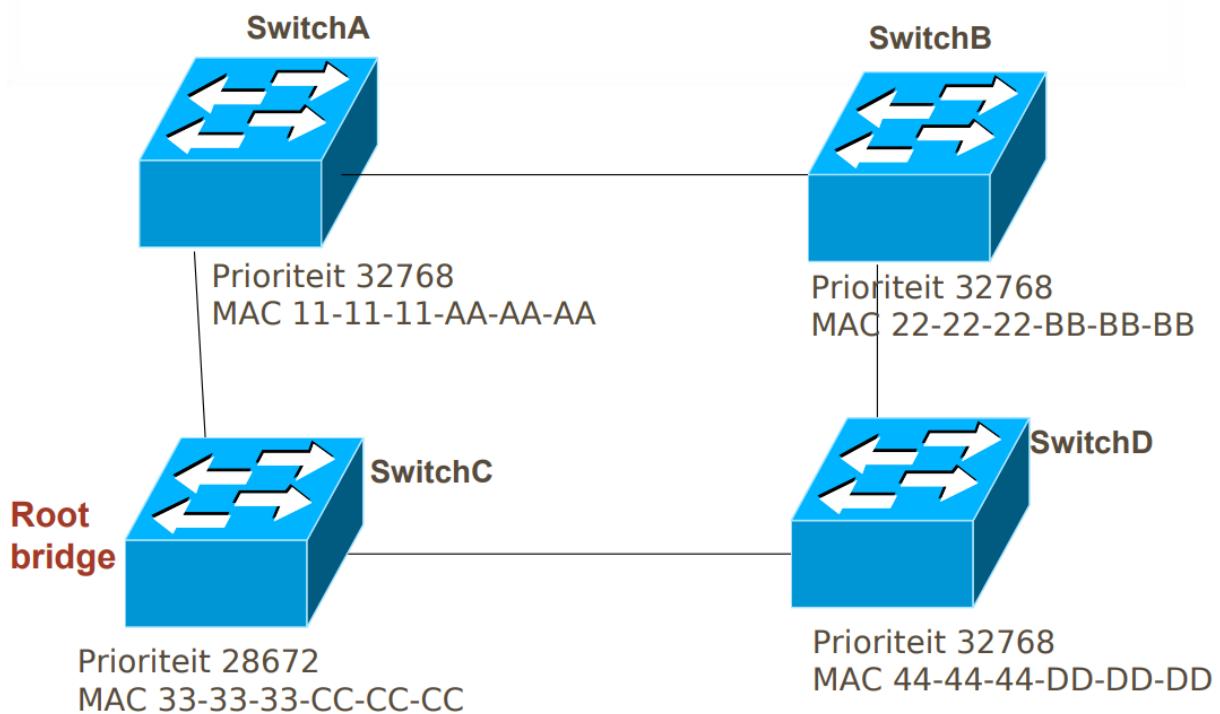
Fase 1

1. Elke switch gaat aan en gedraagt zich als de root. Elke 2 seconden wordt er een **BPDU** uit elke poort gestuurd waarbij de root-BID gelijk is aan de sender-BID
2. Als er een **BPDU** wordt ontvangen met een lagere root-BID dan de eigen BID dan zal het volgende gebeuren
 - a. De switch gaat op alle andere poorten **BPDU** 's sturen met als root-BID de ontvangen BID
 - b. De switch zal geen **BPDU** 's meer genereren (dat doet alleen de root bridge). Wel worden **BPDU** 's van de root bridge doorgestuurd
3. Herhaal dit proces op alle switches totdat er een echte root bridge is
4. De root bridge is nu de enige switch die nieuwe **BPDU** 's verzendt de rest stuurt alleen **BPDU** 's die ze ontvangen van de Root Bridge door naar de volgende switch

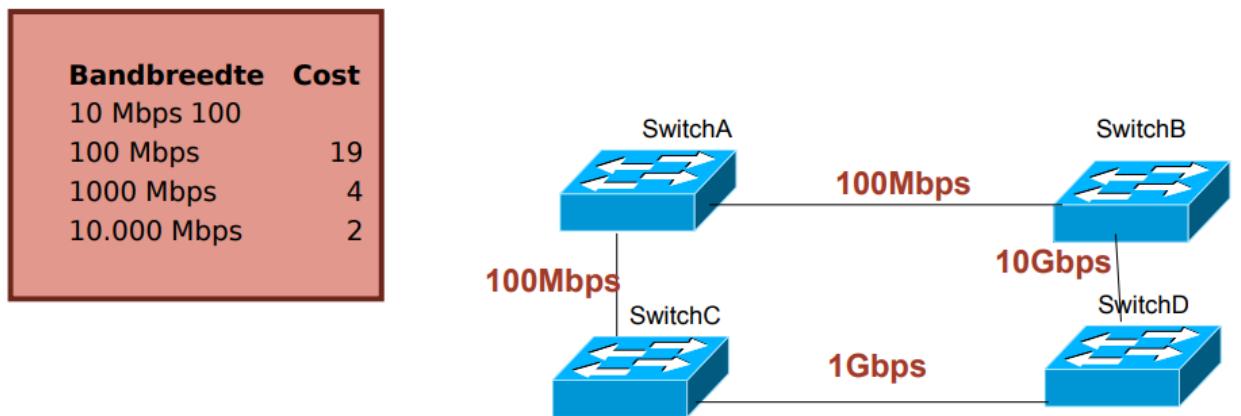
ROOT BRIDGE (=LAAGSTE BID)



ROOT BRIDGE (=LAAGSTE BID) (2)



1. Op de non-root bridges moet gekozen worden welke poorten wel en niet gebruikt moeten worden
2. Dit wordt gedaan door het snelste pad naar de Root Bridge te zoeken d.m.v. *root-path-cost*.
3. De poort met de laagste root-path-cost wordt root-port

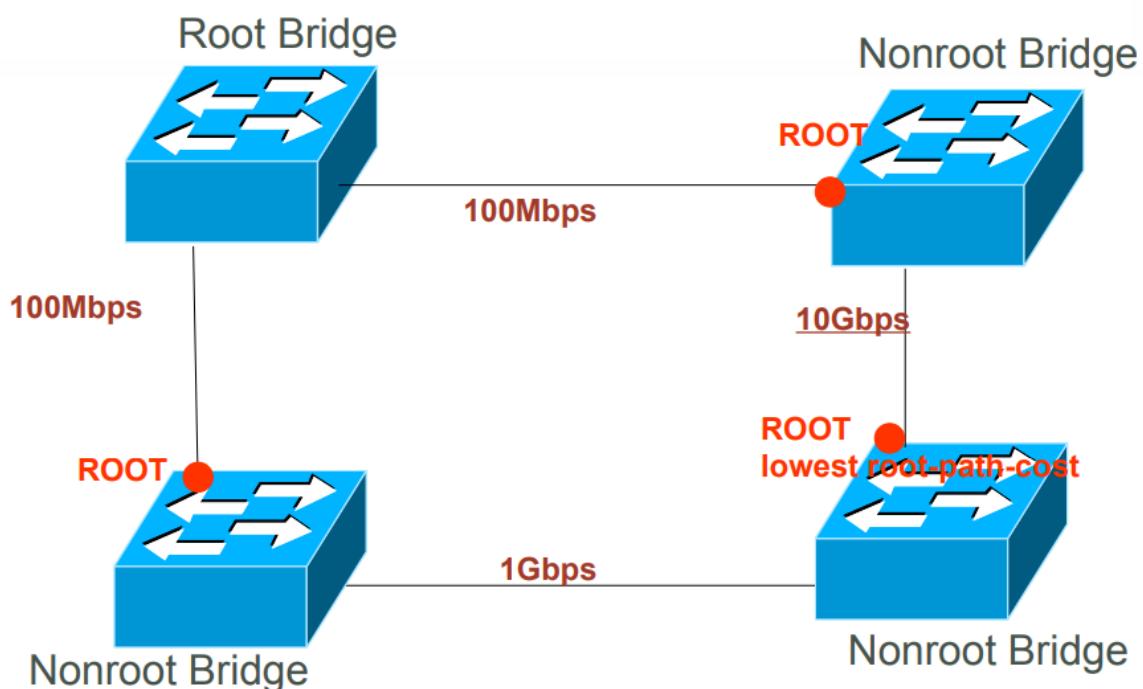


Eventueel zelf in te stellen via:

Switch(config) #**int fa0/1**

Switch(config-if) #**spanning-tree cost 10**

FASE 2: KIEZEN VAN DE ROOT PORTS

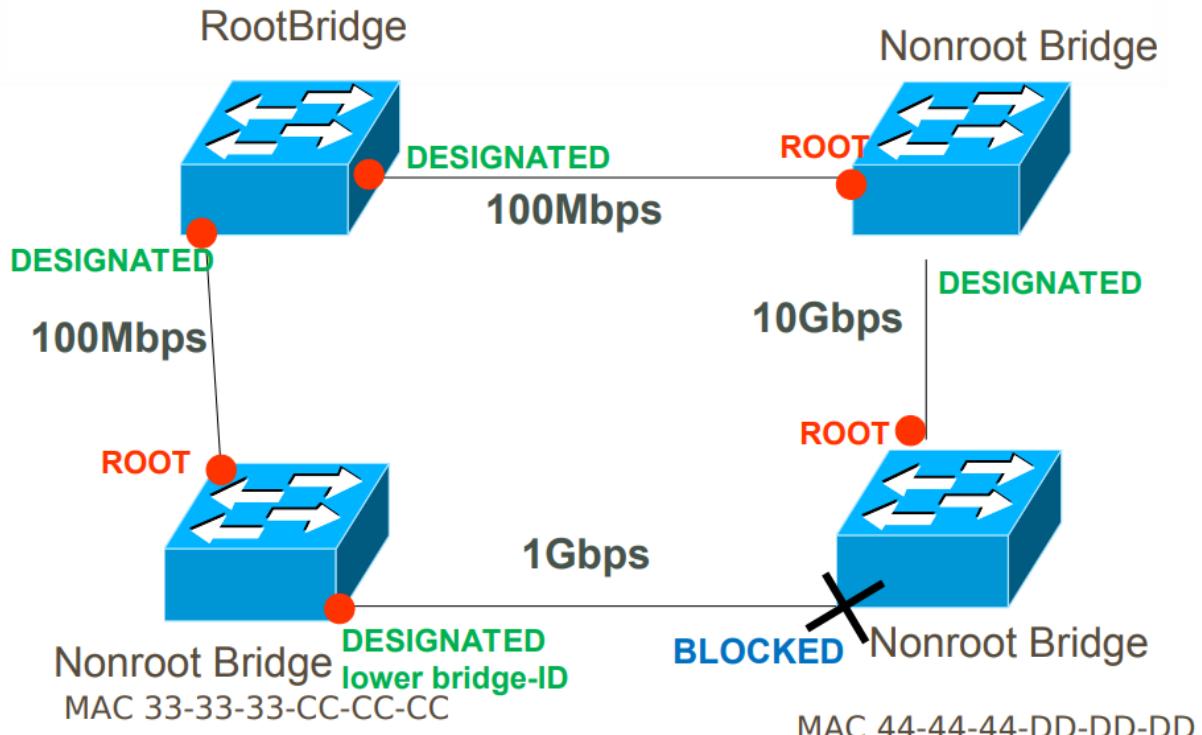


Fase 3

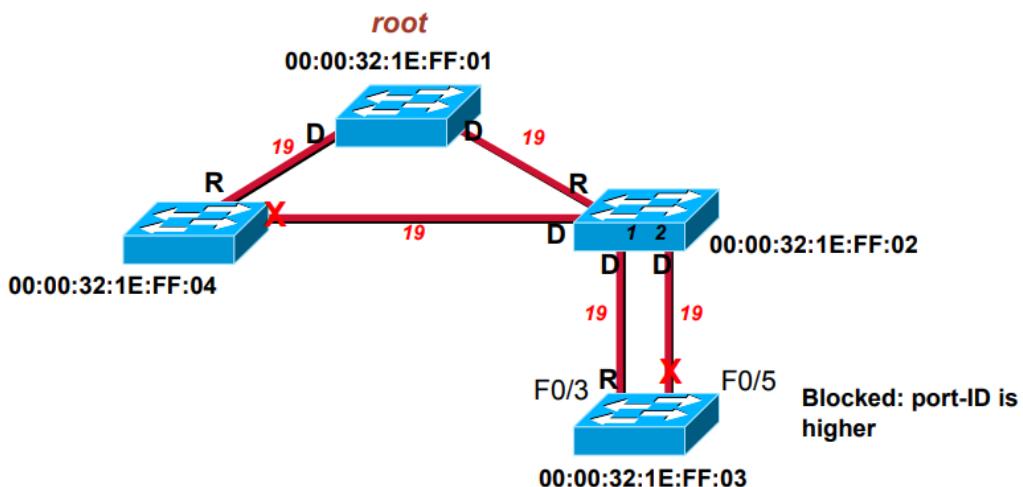
1. Alle poorten van de root bridge zijn *designated ports*

2. Voor de andere ports bepaalt de sender-BID welke port actief blijft (*designated*) en welke niet.
3. Als een switch op twee manieren met één andere switch is verbonden dan zal er een lus zijn, hierbij wordt de port-ID bepaalt om een *designated* port aan te wijzen. De andere port wordt uitgeschakeld

FASE 3: DESIGNATED EN BLOCKED PORTS



FASE 3: DESIGNATED EN BLOCKED PORTS MULTIPLE CONNECTIONS BETWEEN TWO SWITCHES



Port roles

#Handigvoorpdetoe

- Root Port
 - Port met het beste pad naar de root switch
 - Deze poort stuurt verkeer naar de root switch
- Designated Port
 - Alle root poorten
 - Bij de non-root bridges is het de port die van de root afwijst
 - Er kan maar een designated port per segment zijn
- Non-Designated Port
 - Blokkeert frames (**niet** BPDU 's)
 - Voegt geen ingang in aan de MAC-tabel
- Disabled Port
 - Handmatig uitgezet

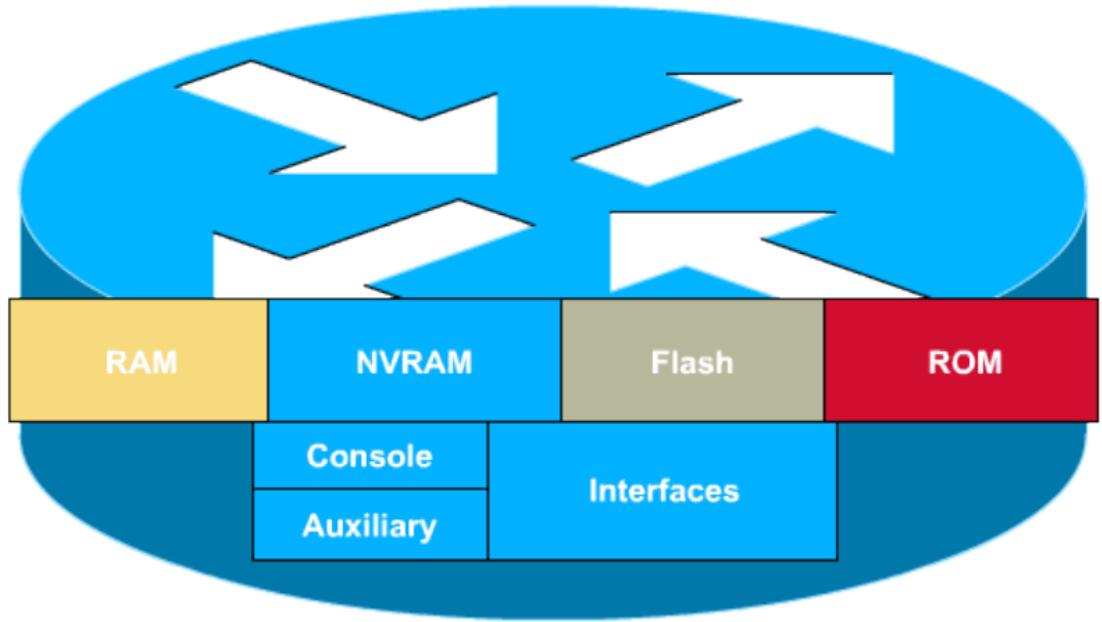
TIMING IN STP

- **Hello Time:** 2 sec (tijd tussen 2 BPDU's).
- **Listening time:** Switches luisteren (15 sec=forward delay) naar BPDU's van buurswitches. 15 sec tijd om STP te laten convergeren.
- **Learning time:** 15 sec (=forward delay). In deze tijd worden MAC-adressen geleerd, maar er wordt nog geen frame geforward.
- **Total forward delay.** De totale tijd dat er gewacht moet worden om een frame te forwarden (30 seconden vanaf begin, of 15 nadat STP geconvergeerd is).
- **Max age:** 20 sec (maximale geldigheid van een ontvangen BPDU)

Week 5 Routing intro, MLS, SVI's

Routers werken op lagen 1,2 en 3 en worden gebruikt om het juiste pad in een netwerk te bepalen. Ook kunnen ze gebruikt worden om: een koppeling tussen twee datalink protocollen te maken; een scheiding van het (inter) netwerk d.m.v. bepaalde regels (bv et blokkeren van gevaarlijk verkeer); en het blokkeren van MAC-broadcasts tussen verschillende netwerkdelen.

OPBOUW VAN EEN ROUTER

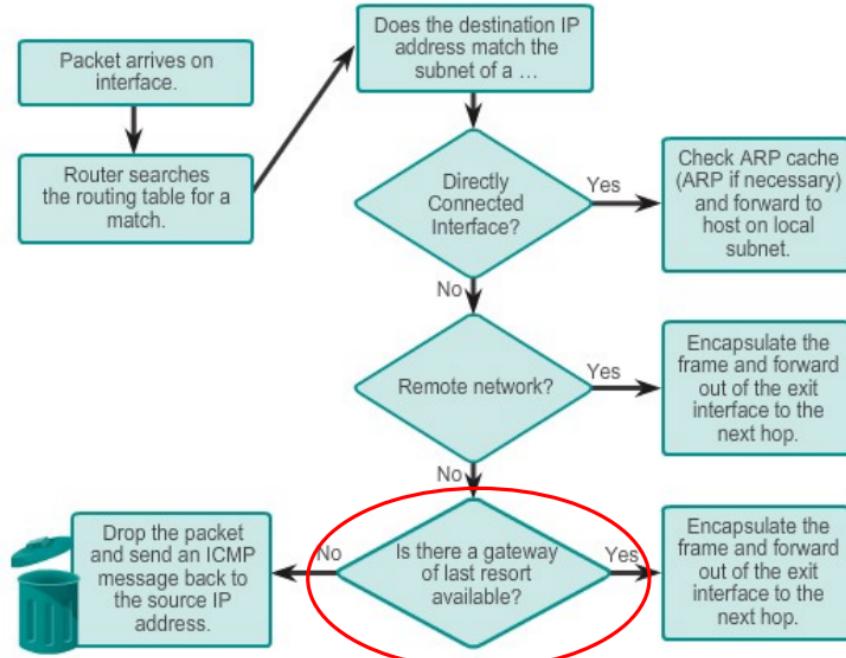


6

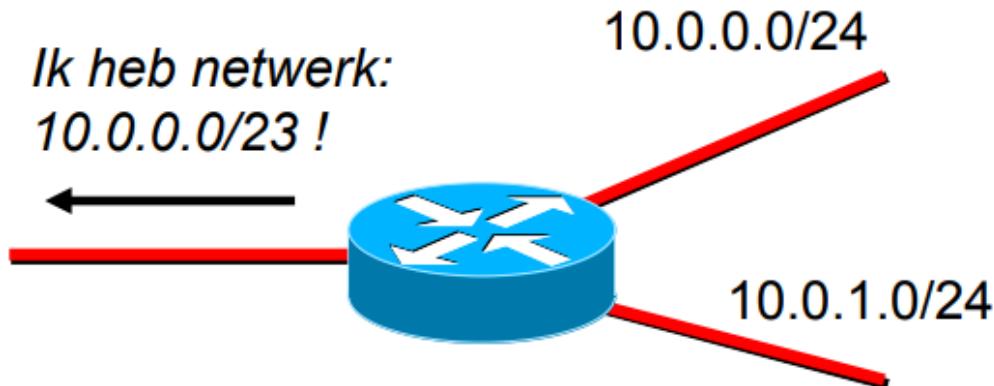
IP Routing

Als een packet aankomt bij een router zal de source MAC-adres worden aangepast naar de MAC-adres van deze router port. Vervolgens zal de destination MAC-adres worden aangepast naar die van de router.

PACKET FORWARDING DECISION PROCESS



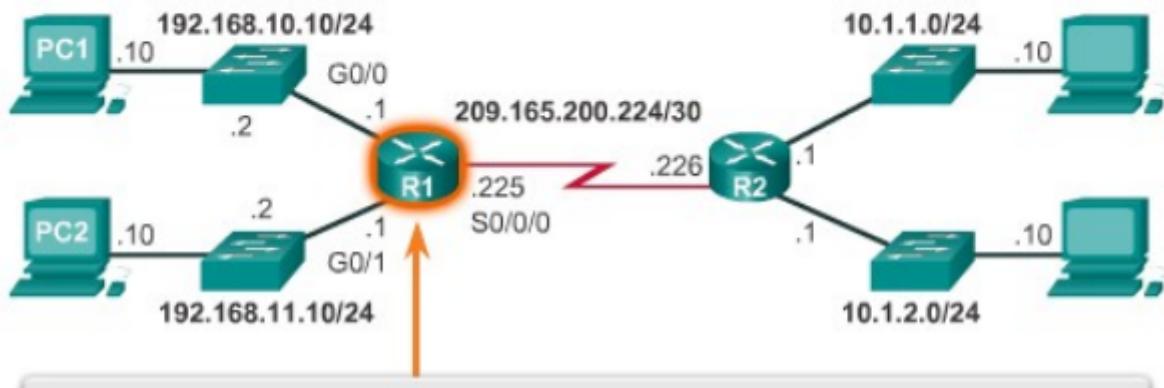
Je kan IP routing gebruiken om te supernetten



Je kan ook een loopback interface aanmaken dit is geen fysieke poort maar wordt als een

software interface gezien. Dit is handig voor het testen van netwerken.

Configure the Loopback0 Interface



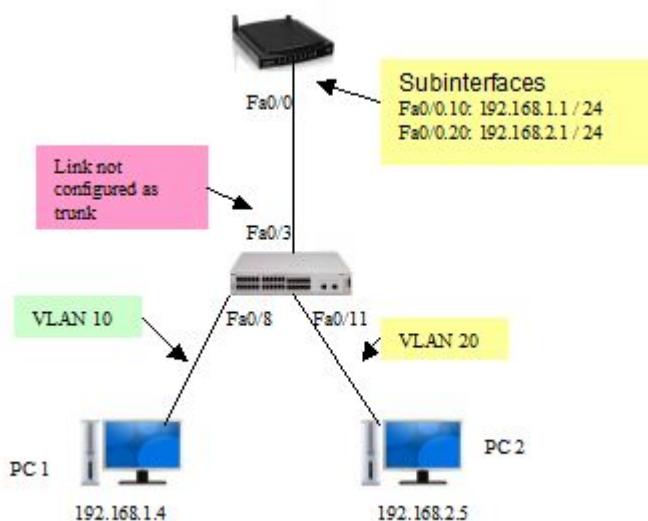
```
R2(config)#interface loopback 0
R2(config-if)#ip address 10.0.0.1 255.255.255.0
R2(config-if)#exit
R1(config)#
*Jan 30 22:04:50.899: %LINK-3-UPDOWN: Interface loopback0,
changed state to up
*Jan 30 22:04:51.899: %LINEPROTO-5-UPDOWN: Line protocol on
Interface loopback0, changed state to up
```

SVI (Switched Virtual Interfaces)

Inter VLAN-Routing

Je kan met gebruik van **Routing-on-a-stick** tussen VLAN's routeren

Configuratie voorbeeld router "On a stick"



Vergeet niet om de no shutdown command toe te voegen

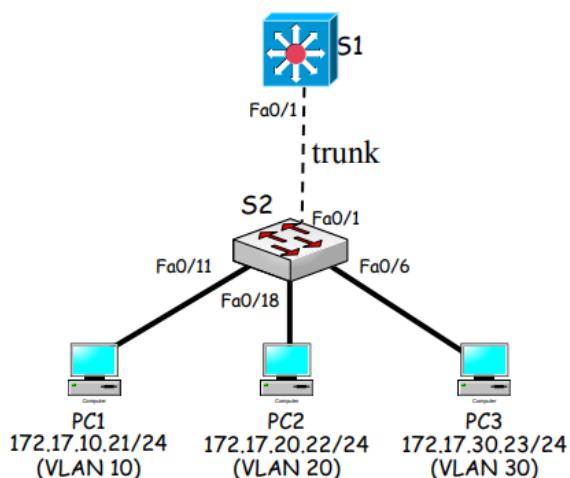
```
$ no shutdown
```

Met dit netwerk kun je zowel pc 1 als pc 2 pingen

Maar dit kan ook met gebruik van een MLS (Multi Layer Switch), dit wordt gedaan met gebruik van SVI. Deze SVI is een virtuele interface voor een VLAN naar een laag 3 routing systeem. SVI wordt 1 op 1 met een VLAN ingesteld er kan dus maar een SVI per VLAN zijn.

- SVIs allow traffic to be routed between VLANs by providing a default gateway for the VLAN.
- Provide Layer 3 IP connectivity to the switch.
- Support routing protocol.
- SVIs are faster than router-on-a-stick, because the packets are hardware-switched and routed.
- Not limited to one link, Layer 2 EtherChannels can be used between the switches to get more bandwidth.
- No need for external links from the switch to the router for routing.
- Latency is much lower, because it does not need to leave the switch
- An SVI aka Routed VLAN Interface (RVI) by some vendors.

MULTILAYER SWITCH SVI CONFIGURATION



Configure SVI Addresses:

```
S1(config)#int vlan 10
S1(config-if)#ip address 172.17.10.1 255.255.255.0
S1(config-if)#int vlan 20
S1(config-if)#ip address 172.17.20.1 255.255.255.0
S1(config-if)#int vlan 30
S1(config-if)#ip address 172.17.30.1 255.255.255.0
```

```
S1(config)#ip routing
```

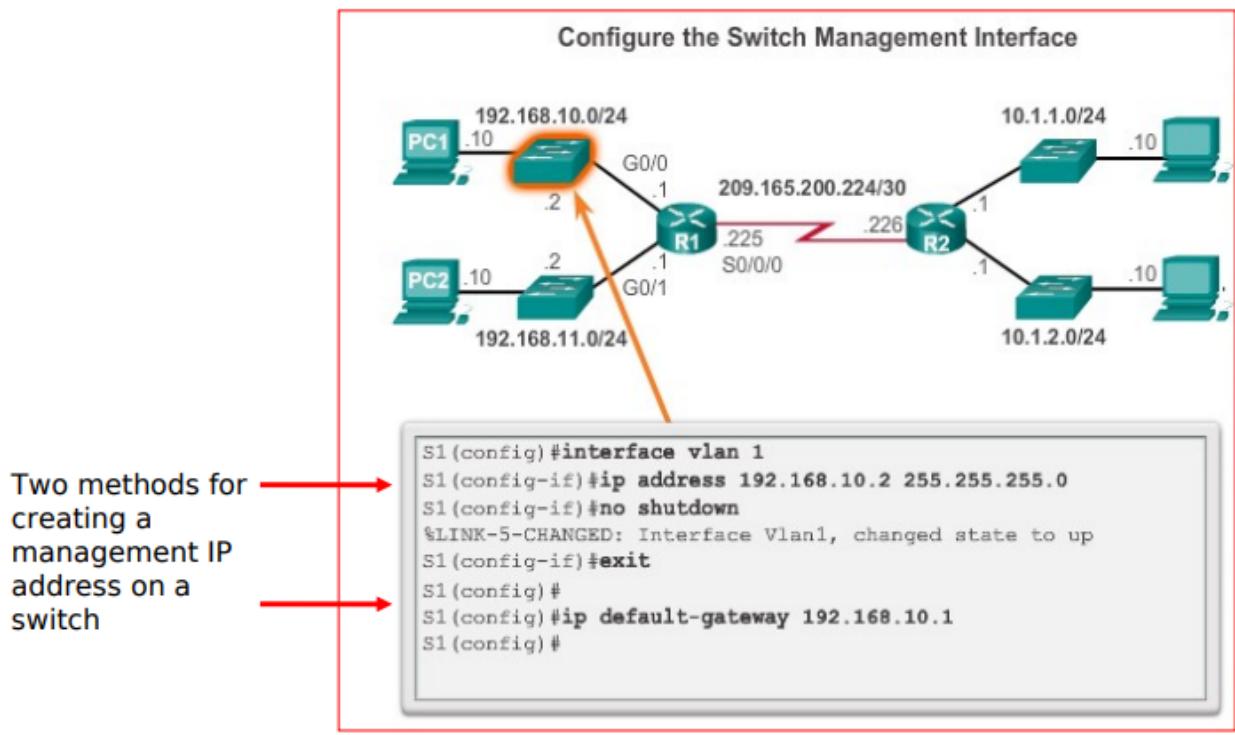
```
S1#show ip route
```

172.17.0.0/24 is subnetted, 3 subnets

```
C    172.17.10.0 is directly connected, Vlan10
C    172.17.20.0 is directly connected, Vlan20
C    172.17.30.0 is directly connected, Vlan30
```

Hierboven is een configuratie van een SVI te zien, het is belangrijk dat je altijd *ip routing* aanzet anders werkt dit niet.

Je kan SVI ook gebruiken voor remote management met SSH



Er zijn 3 factoren die bepalen of dat een SVI aanstaat

1. De VLAN bestaat en is *actief* in de VLAN database op de switch
2. De VLAN interface *bestaat* en is niet administratively down
3. Ten minste een laag 2 (acces of trunk) port bestaat op de switch en heeft een verbinding met de *up state* van deze VLAN, en deze staat in de spanning-tree *forwarding* state op de VLAN

Als dit alles waar is kan je de SVI controleren als volgt

```
S1# show interfaces vlan 20
Vlan20 is up, line protocol is up
Hardware is Ethernet SVI, address is 00D.588F.B604 (bia 00D.588F.
Internet address is 10.1.20.1/24
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drop
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
```

Voor en Nadelen

Voordelen

1. Het is veel *sneller* dan *routing-on-a-stick* omdat alles via *hardware* wordt gerouteerd.
2. Er is *geen* nood voor *externe* verbindingen van de switch naar de router voor routering.
3. Ook kunnen laag 2 *EtherChannels* gebruikt worden om meer bandbreedte te gebruiken tussen switches
4. *Latency* is veel lager omdat de switch niet verlaten wordt

Nadelen

Je moet een *laag 3 switch* hebben om Inter-VLAN routing te gebruiken, en deze zijn erg duur.

Op een multi layer switch kan gebruik gemaakt worden van

1. Laag 2 switchpoorten
2. Laag 3 SVI's
3. Laag 3 ports(routed ports)

Routed ports

Een routed port is een fysieke poort met laag 3 toepassingen. Deze port is niet verbonden met een VLAN, en heeft een soortgelijk interface aan dat van een router. Laag 2 functionaliteit is verwijderd en daarom kan deze niet geconfigureerd worden.

SUMMARIZING MLS LAYER 3 CONFIGURATION COMMANDS

- Enable L3 routing on an MLS

```
MLS(config)#ip routing
```

- Create a routed port

```
MLS(config)#interface Fa0/7
MLS(config-if)#no switchport
MLS(config-if)#ip address 10.10.10.1 255.255.255.0
MLS(config-if)#no shutdown
```

- Create an SVI

```
MLS(config)#interface vlan 10
MLS(config-if)#ip address 10.10.20.1 255.255.255.0
MLS(config-if)#no shutdown
```

Make sure that the vlan exists, using command `MLS(config)#vlan 10`

Week 6 VTP (VLAN Trunk Protocol)

VTP Maakt het makkelijker om VLAN databases tussen switches te beheren. Dit is handig omdat naarmate je meer switches gebruikt in een netwerk het moeilijker wordt om VLANs en trunks te beheren.

VTP wordt gebruikt om **automatisch** informatie van VLANs tussen switches uit te wisselen

Voordelen en nadelen

Voordelen

VTP maakt het beheren van de VLAN database tussen meerdere switches makkelijker en dus zal de VLAN configuratie constant blijven door het gehele netwerk. VTP log-melding kan worden gegeven als een VLAN wordt toegevoegd in het netwerk

Nadelen

Je moet nog steeds poorten toewijzen aan de VLANs

Revisienummer

Het configuratie-revisienummer is een 32-bits nummer dat het revisieniveau voor een VTP-frame aangeeft.

Het standaard configuratienummer voor een switch is nul.

Elke keer dat een VLAN wordt toegevoegd of verwijderd, wordt het configuratie-revisienummer verhoogd. Elk VTP-apparaat houdt het revisienummer van de VTP-configuratie bij dat eraan is toegewezen.

- Note: Een wijziging van de VTP-domeinnaam verhoogt het revisienummer niet. In plaats daarvan wordt het revisienummer teruggezet naar nul.

Probleem met revisienummer

- Als een nieuwe switch wordt toegevoegd aan het netwerk met een hoger revisienummer, wordt de server bijgewerkt met de VLAN-database van de nieuwe switch
- Hierdoor kan het netwerk onbruikbaar worden omdat de verkeerde VLAN-info door deze nieuwe *server* wordt doorgegeven

Pruning

Met pruning wordt de capaciteit van het netwerk verhoogt doordat je VTP-info alleen door trunk verbindingen, die gebruikt worden om betreffende hosts te bereiken, stuurt.

Switches communiceren dan over de VLANs die aan het eind van de trunk bereikbaar zijn
Je schakelt pruning in met het commando `vtp pruning`

Modes

#Handigvoorpdetoets

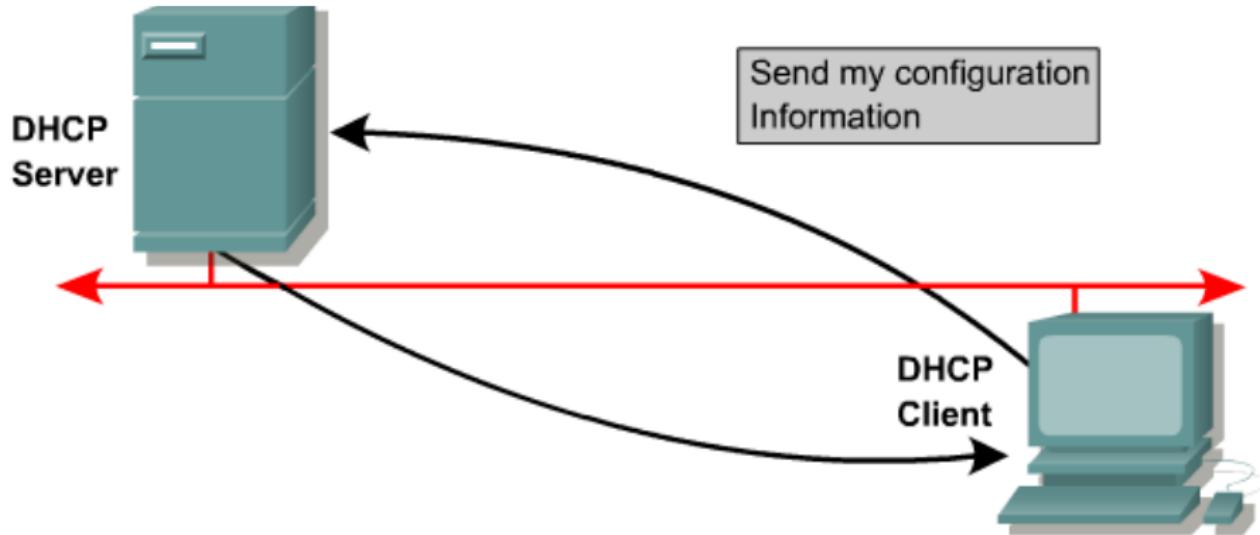
Er zijn 3 VTP operating modes

- **Client**
 - Kan geen VLANs creëren, verwijderen en veranderen
 - Krijgt VLAN informatie van server
- **Server**
 - Kan VLANs creëren, verwijderen en veranderen
- **Transparant**
 - Kan lokaal VLANs creëren, verwijderen en veranderen
 - Stuurt VTP-berichten door

DHCP

Voor het toekennen van **dynamische** adressen wordt **DHCP** gebruikt

- Schaalbaar
- Simpel
- Een router kan als DHCP server worden geconfigureerd
- Dure serverapparatuur is niet nodig



Here is Your Configuration:

- IP Address: 192.204.18.7
- Subnet Mask: 255.255.255.0
- Default Routers: 192.204.18.1, 192.204.18.3
- DNS Servers: 192.204.18.8, 192.204.18.9
- Lease Time: 5 days

Werking

- DHCP levert IP-gerelateerde informatie van een (DHCP) sever aan clients deze gegevens worden tijdelijk als lease verstreken.
- Hierdoor kunnen later andere clients het IP-adres krijgen
- Naast het IP-address en netmask levert DHCP ook : default-gateway_(router), DNS-adres, WINS-server, lease-duur, domain-name, ...

Configuratie

```

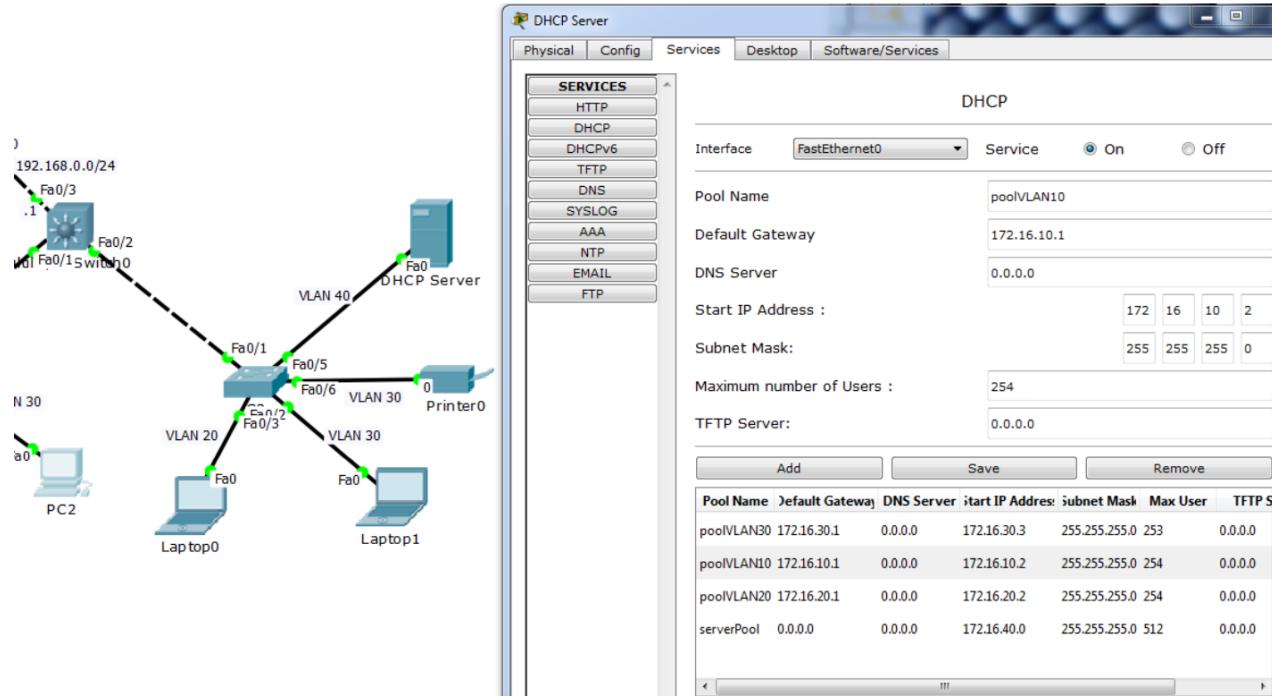
Router(config)#ip dhcp pool subnet12
Router(dhcp-config)#network 172.16.12.0 255.255.255.0
Router(dhcp-config)#default-router 172.16.12.254
Router(dhcp-config)#dns-server 172.16.1.2
Router(dhcp-config)#netbios-name-server 172.16.1.3
Router(dhcp-config)#domain-name foo.com

```

~~Router(config)#ip dhcp excluded-address
ip-address [end-ip-address]~~

~~Router(config)#ip dhcp excluded-address 172.16.1.1 172.16.1.10
Router(config)#ip dhcp excluded-address 172.16.1.254~~

Voor statische ip -adressen (servers, printers)

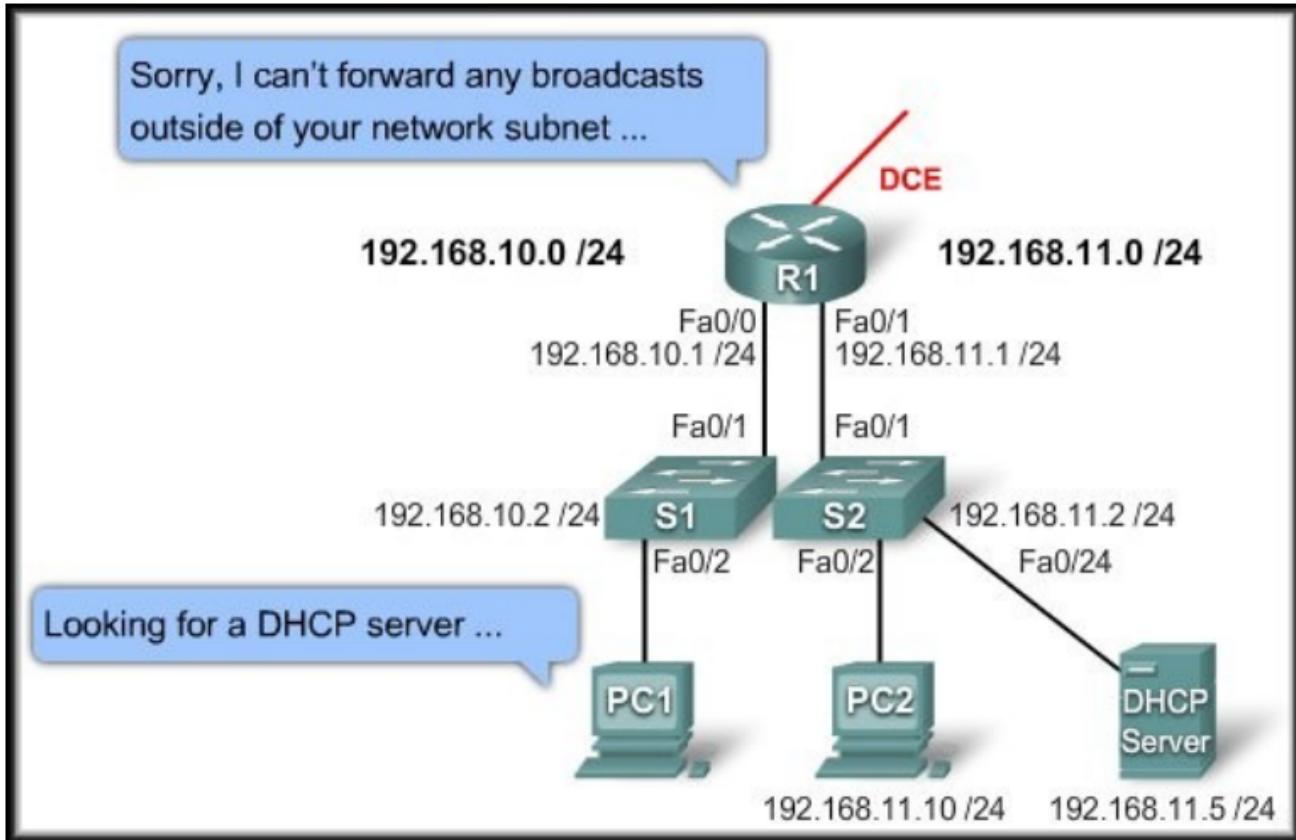


Relay

Probleem met DHCP

Als een DHCP server in een ander netwerk staat dan worden discover messages van de

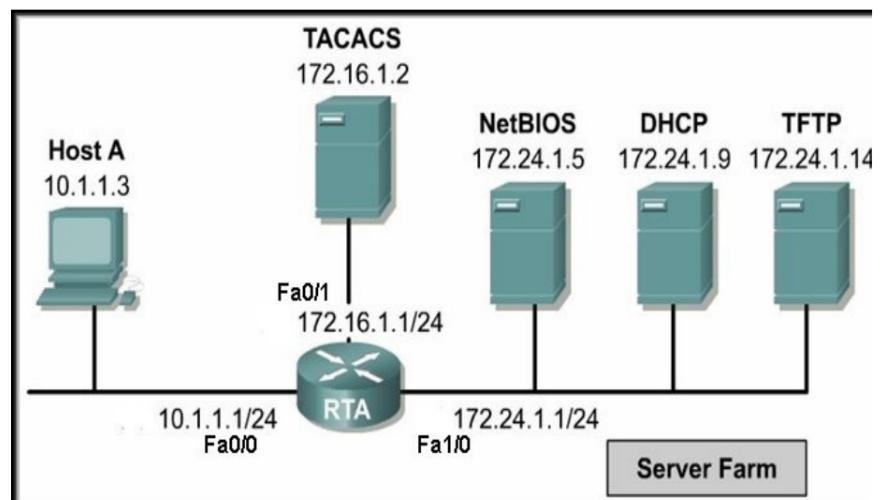
host niet ontvangen.



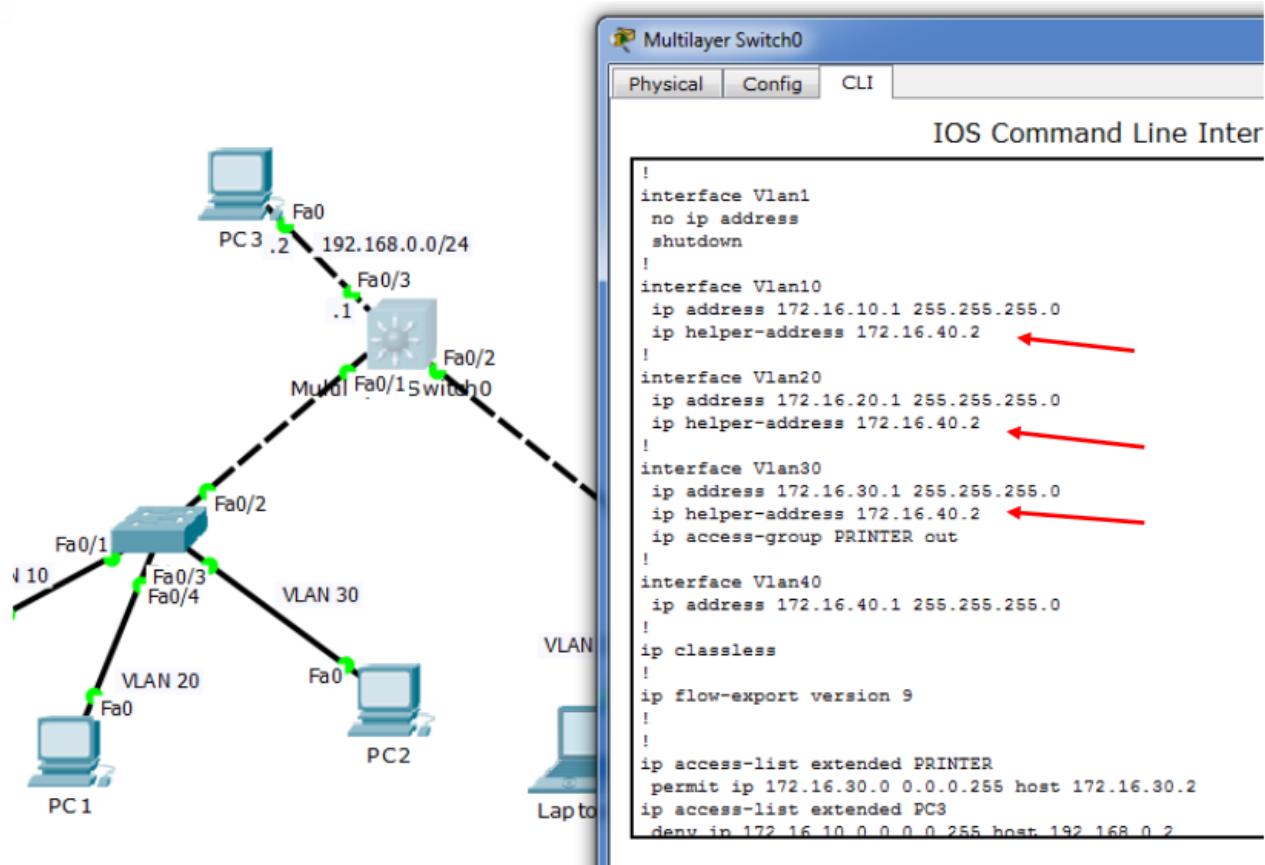
Oplossing: IP Helper Address

Door een **helper address** te configureren op tussenliggende routers zullen de DHCP broadcasts doorgestuurd worden naar de betreffende servers.

- RTA(config)#interface fa0/0
RTA(config-if)#ip helper-address 172.24.1.9



Deze helper adressen kunnen ook op SVI gebruikt worden



DHCP is niet het enige protocol waar broadcasts gebruikt worden

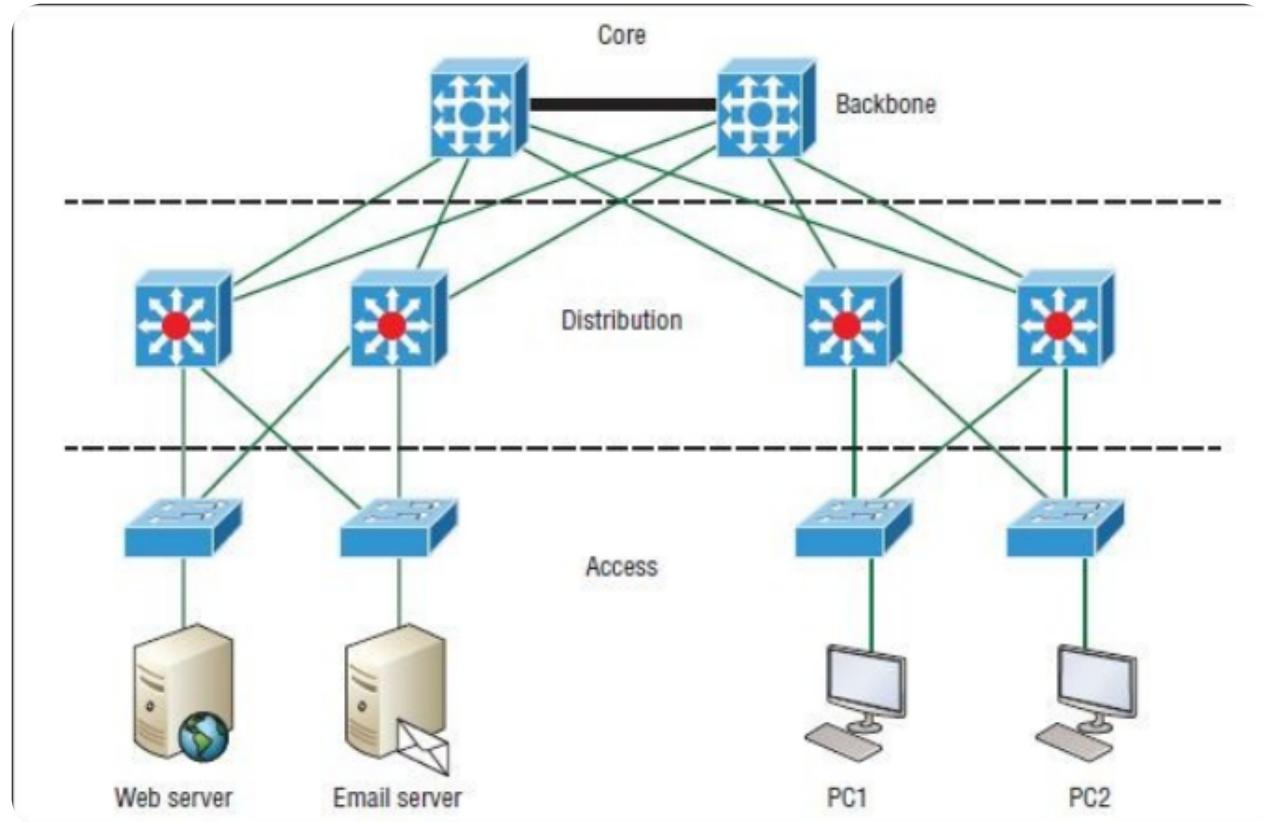
Service	Port
Time	37
TACACS	49
DNS	53
BOOTP/DHCP server	67
BOOTP/DHCP client	68
TFTP	69
NetBIOS name service	137
NetBIOS datagram service	138

Voor info over de lagen zie

Lagen

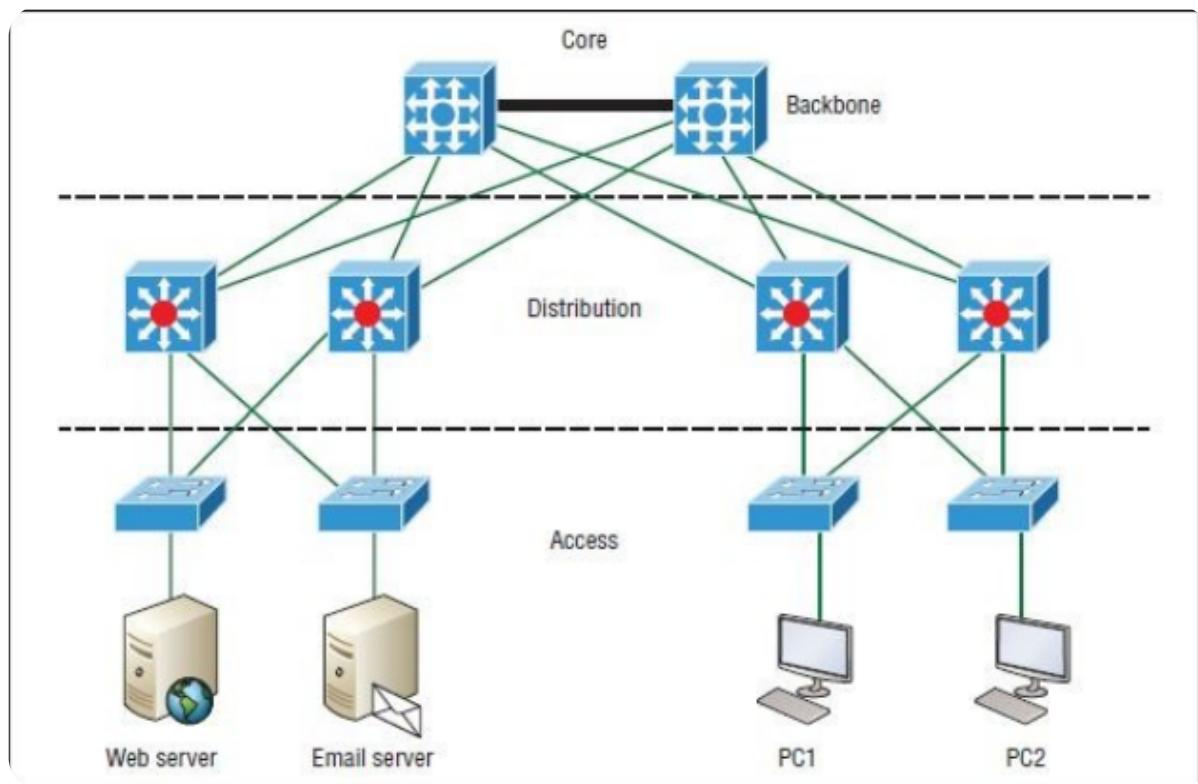
Acces layer

De laag met de end devices (PC's, servers, printers, wireless acces points)



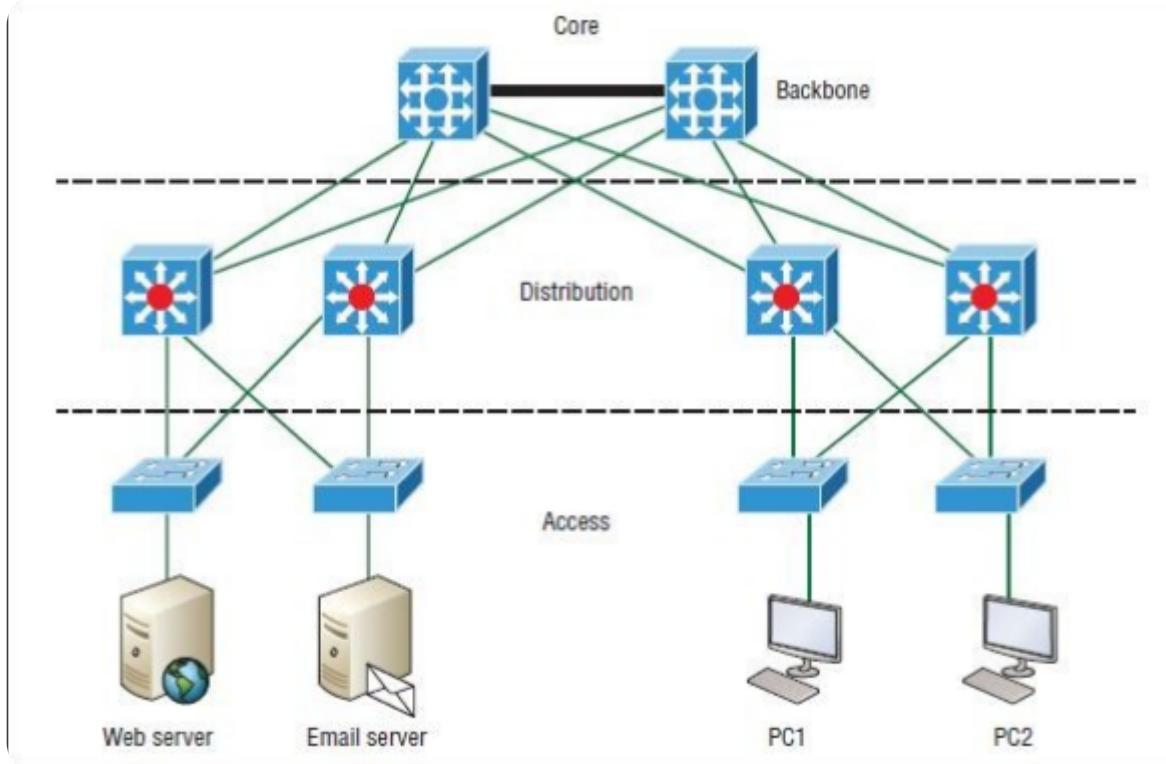
Distribution layer

- Aggregatie van/naar de acces layer
- Hier komen de meeste security en routing regels
- Verdeelt doorgaans de acces laag in VLANs

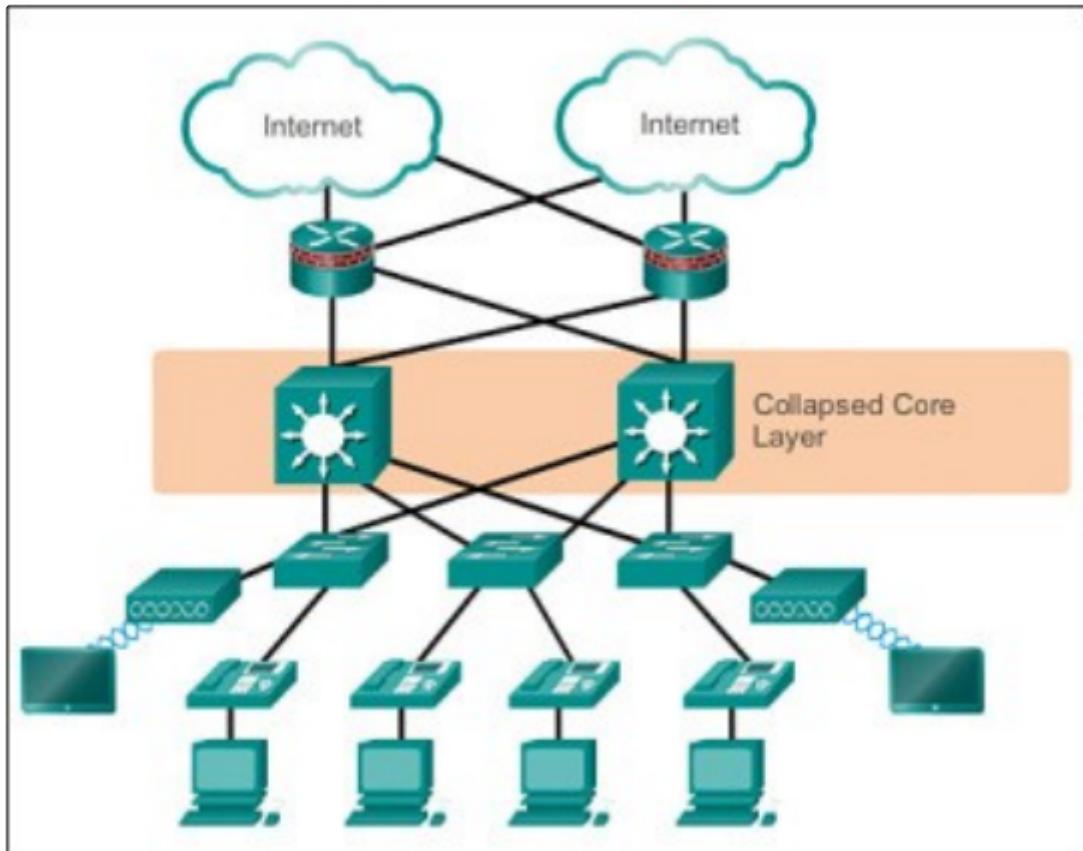


Core Layer

- High speed backbone van het netwerk Hoge availability en redundant uitgevoerd
 - Moet snel, veel data kunnen forwarden
 - Bij kleinere netwerken, collapsed model (Core and distribution samengevoegd)



Collapsed core



De meeste bedrijven zullen deze structuur gebruiken, een extra core layer zie je alleen in

Redundantie

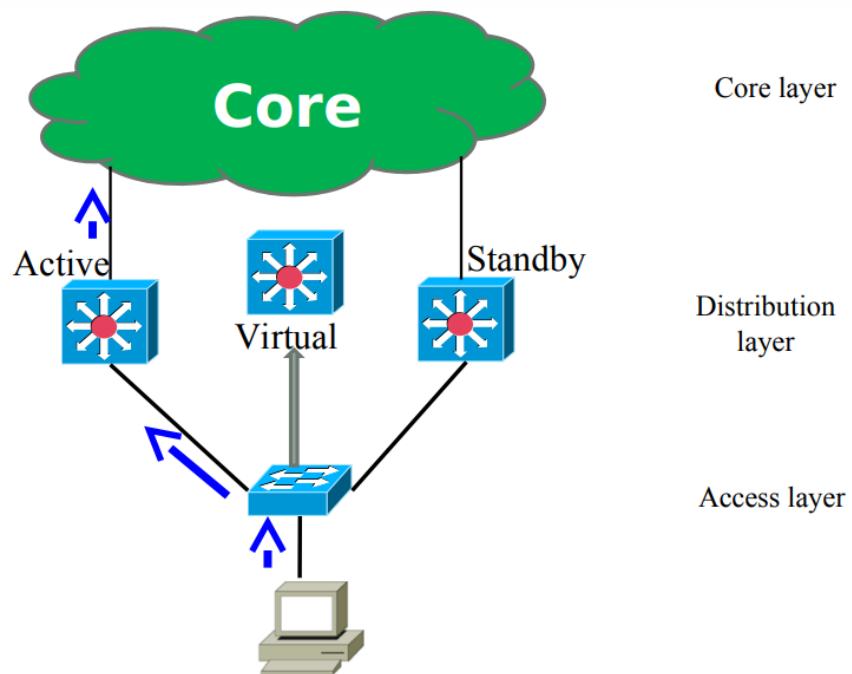
First Hop redundancy Protocols

1. HSRP (Hot standby Routing Protocol)
 - Cisco proprietary
2. VRRP (Virtual Redundancy Protocol)
 - IEEE Standard
 - Similar to HSRP
 - Niet daadwerkelijk tussen routers gebruiken
3. GLHP (Gateway Load Balancing Protocol)
 - Cisco proprietary
 - Met load balancing

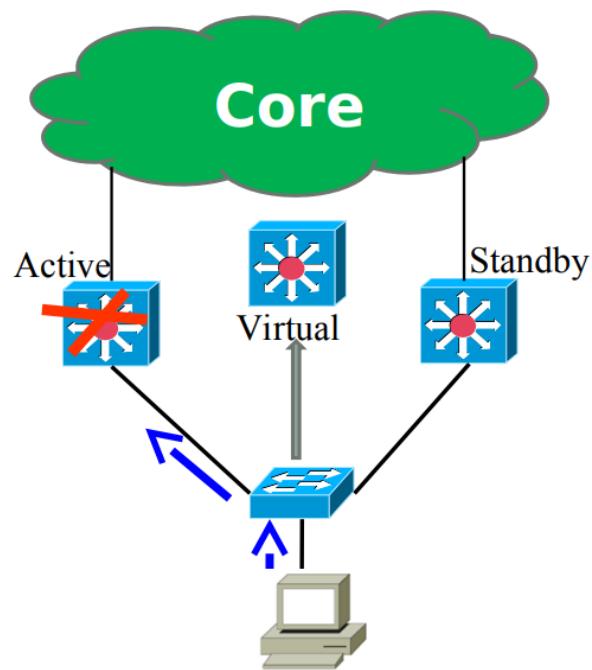
HSRP (Hot Standby Routing Protocol)

- RFC 2281
- Provides network redundancy for IP networks
- Automatic recovery from first-hop failures.
- Use UDP port 1985
- Multicast address 224.0.0.2 (version 1); 224.0.0.102 (version 2)
- TTL=1

PRINCIPLE HSRP (1)

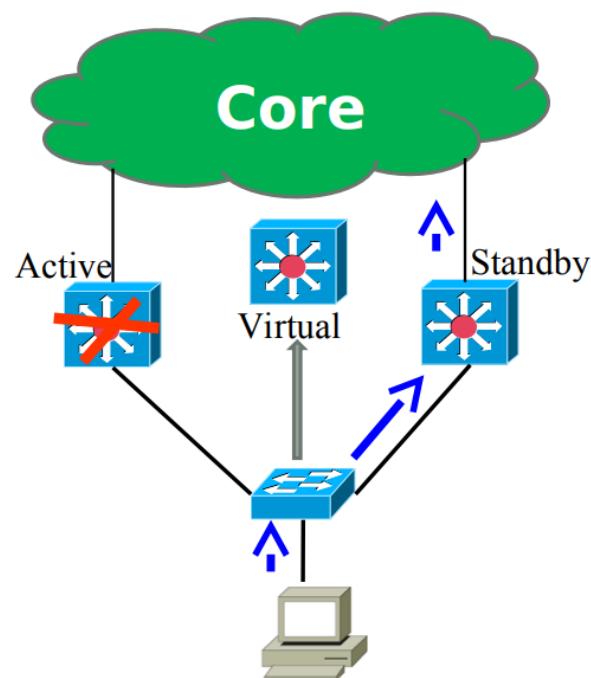


PRINCIPLE HSRP (2)



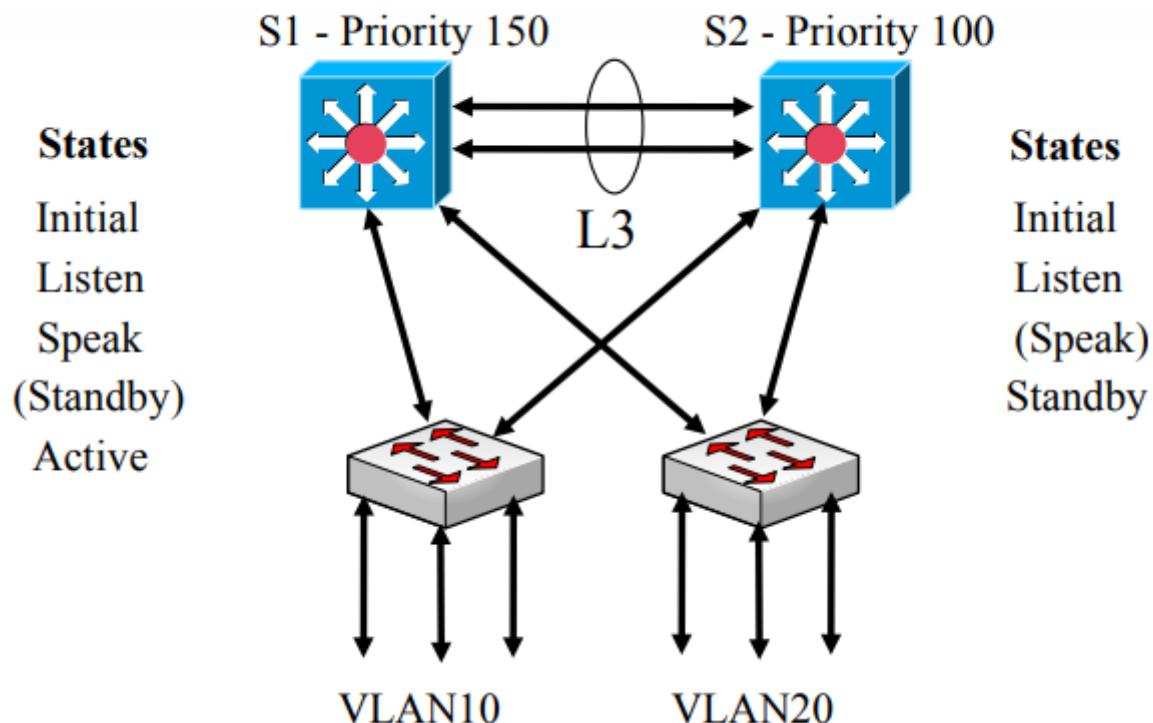
46

PRINCIPLE HSRP (3)



- **Active router** - Doet het doorsturen van datapakketten en verzendt hallo-berichten naar andere routers om hen op de hoogte te stellen van de status ervan?
- **Standby router** - Bewaakt de status van de actieve router en begint snel met het doorsturen van pakketten in het geval van een actieve routerstoring.
- **Virtual Router** - **Bestaat niet!** Vertegenwoordigt een consistent beschikbare router met een IP-adres en een MAC-adres voor de hosts op een netwerk.
- **Other routers** - Houd HSRP-hallo-berichten in de gaten, maar reageer niet. Functioneren als normale routers die naar hen verzonden pakketten doorsturen, maar geen pakketten doorsturen die zijn geadresseerd aan de virtuele router.

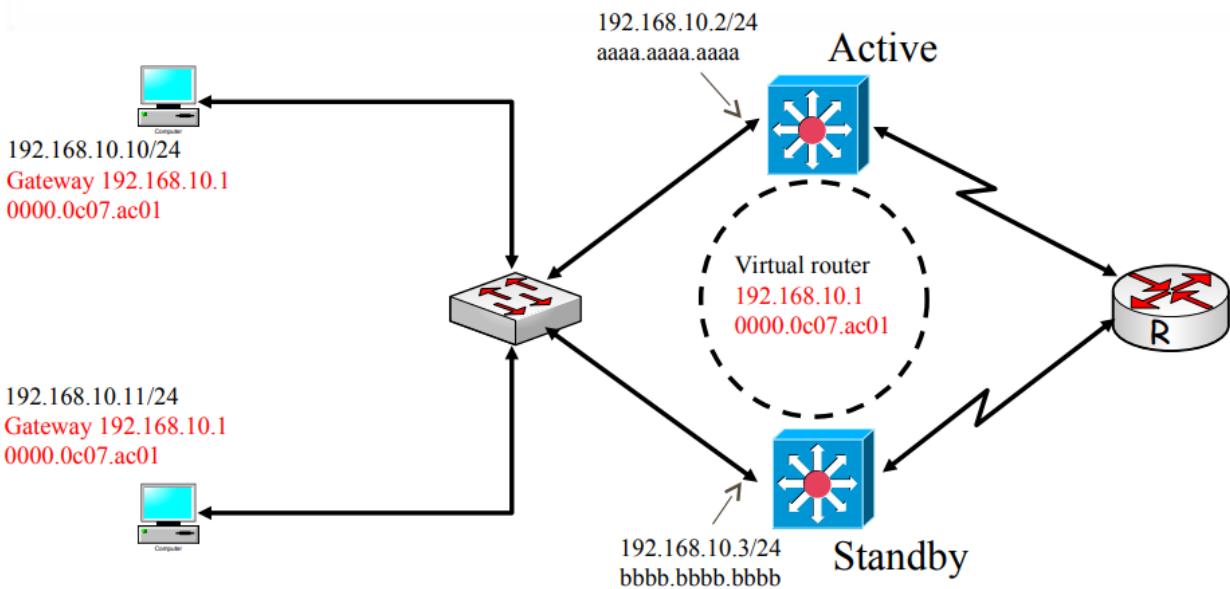
Progression



Beide switches starten in de Initial status, verwerken de configuraties en gaan dan verder naar de Listening status, terwijl ze HSRP-hello's op het netwerk verwachten:

- Als hello's na een time-out niet worden ontvangen, gaat een switch naar de speak status en kiest actief een nieuw actief en standby-apparaat door naar elkaars hello-pakketten te kijken om te bepalen welke router welke rol moet aannemen.
- Als hello's wordt ontvangen voordat de time-out is verstreken, blijft de switch in de listening status en wordt afhankelijk van de prioriteit naar Actief of Stand-by verplaatst.

Operation



- The virtual router is a virtual IP and MAC address pair that end devices have configured as their default gateway.
- The active router processes all packets and frames sent to the virtual router address.
- The HSRP standby router monitors the operational status of the HSRP group and quickly assumes packet-forwarding responsibility if the active router becomes inoperable.

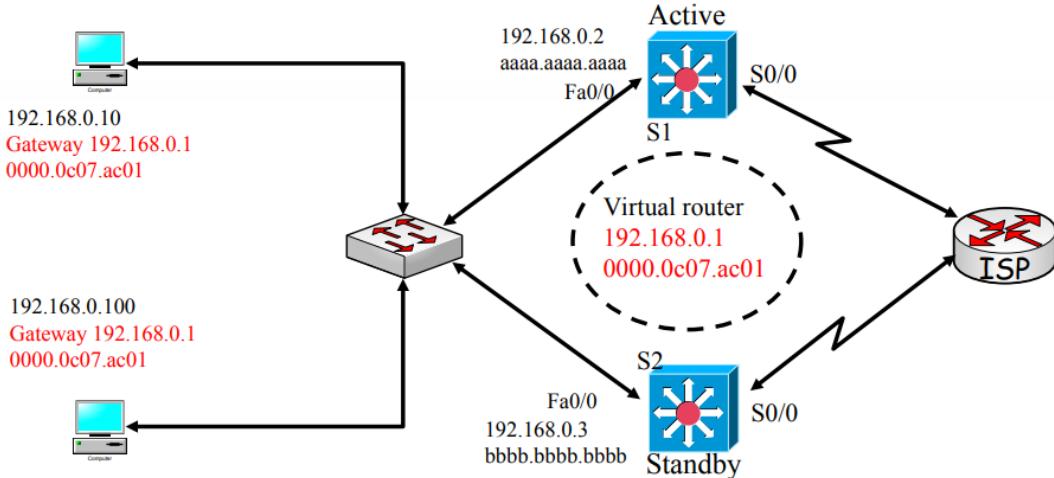
- Wanneer de standby-router geen hello-berichten meer ontvangt van de actieve router, wordt deze een actieve router.
- Omdat de hosts een virtueel IP- en MAC-adres gebruiken, zien ze weinig tot geen onderbreking van de service.
- In het zeldzame geval dat zowel de actieve als de standby-routers falen, zullen alle andere routers in de groep worden gekozen voor de actieve en standby-rollen;
- De router met het hoogste IP-adres op de HSRP-interface wordt de actieve router, tenzij er een HSRP-prioriteit is geconfigureerd.

MAC-adres

0000.0c06.ac01

- **Vendor ID (Vendor Code)** - De eerste 3 bytes van het MAC-adres , als deze onbekend is wordt het gelijk gezet aan 0000.0c
- **HSRP Code (HSRP standard virtual MAC address)** - De volgende 2 bytes van het MAC-adres *zijn altijd 07.ac*
- **Group ID (HSRP group number in hex)** - De laatste byte van het MAC-adres

Configuratie



Active

```
S1(conf)#int fa0/0
S1(conf-if)#ip address 192.168.0.2 255.255.255.0
S1(conf-if)#standby 1 ip 192.168.0.1
S1(conf-if)#standby 1 priority 150
S1(conf-if)#standby 1 preempt
S1(conf-if)#standby 1 preempt delay minimum 225
```

Standby

```
S2(conf)#int fa0/0
S2(conf-if)#ip address 192.168.0.3 255.255.255.0
S2(conf-if)#standby 1 ip 192.168.0.1
S2(conf-if)#standby 1 priority 100
S2(conf-if)#standby 1 preempt
```

Preempt wordt gebruikt om een wijziging in de actieve/passieve modus te forceren na een wijziging van de prioriteitswaarde, bijvoorbeeld nadat de actieve router is uitgevallen (en de stand-by actief is geworden) en is gerepareerd, neemt het de actieve rol weer aan. Het **delay-commando** zorgt ervoor dat de router wacht voordat deze weer actief wordt (standaard = 0). In dit geval wacht hij minimaal 225 seconden. Nuttig om te wachten tot de routeringstabellen zijn bijgewerkt.

```
SwitchA(config)#interface vlan 10
SwitchA(config-if)#ip address 10.1.10.2 255.255.255.0
SwitchA(config-if)#standby 10 ip 10.1.10.1
SwitchA(config-if)#standby 10 priority 110
SwitchA(config-if)#standby 10 preempt
```

```
SwitchB(config)#interface vlan 10
SwitchB(config-if)#ip address 10.1.10.3 255.255.255.0
SwitchB(config-if)#standby 10 ip 10.1.10.1
SwitchB(config-if)#standby 10 priority 90

SwitchB#show standby [brief]
```

IP virtual router

Becomes the active router

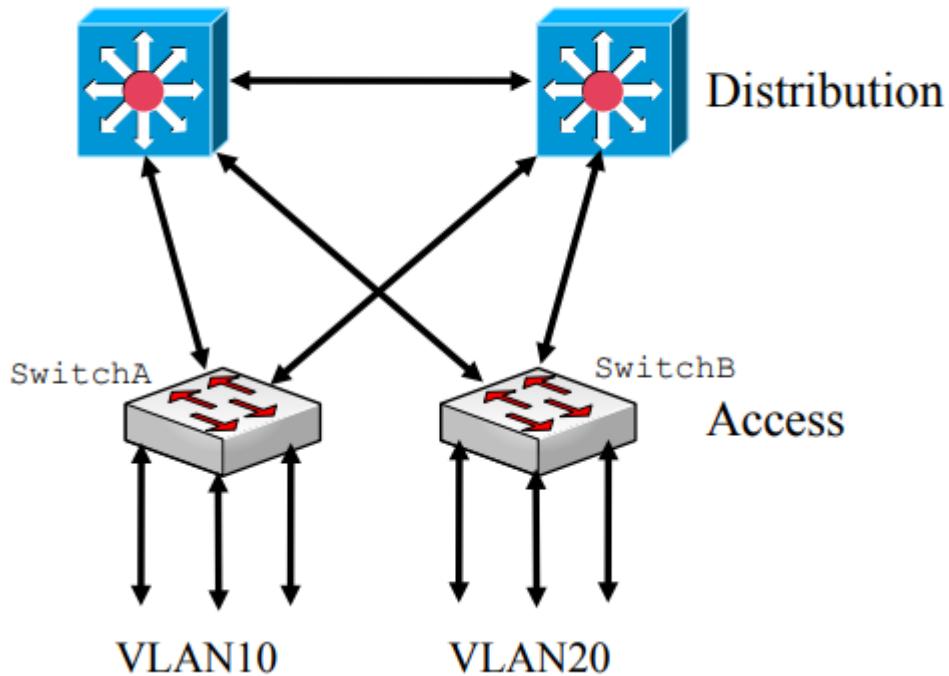
Take active role back when possible

Spanning Tree (HSRP)

De STP **root** van elke VLAN zal gelijk moeten zijn aan de **active** HSRP uitvoering. Op deze manier is er ook load balancing.

HSRP Active VLAN 10
STP Root VLAN 10

HSRP Active VLAN 20
STP Root VLAN 20



Assign STP priorities
per VLAN

```
SwitchA(config)# spanning-tree vlan 10 root primary  
SwitchA(config)# spanning-tree vlan 20 root secondary
```

Assign different HSRP
priorities per VLAN SVI

```
SwitchA(config)# interface vlan 10  
SwitchA(config-if)# ip address 10.1.10.2 255.255.255.0  
SwitchA(config-if)# standby 10 ip 10.1.10.1  
SwitchA(config-if)# standby 10 priority 110  
SwitchA(config-if)# standby 10 preempt
```

```
SwitchA(config)# interface vlan 20  
SwitchA(config-if)# ip address 10.1.20.2 255.255.255.0  
SwitchA(config-if)# standby 20 ip 10.1.20.1  
SwitchA(config-if)# standby 20 priority 90  
SwitchA(config-if)# standby 20 preempt
```

```

SwitchB(config)#spanning-tree vlan 20 root primary
SwitchB(config)#spanning-tree vlan 10 root secondary

SwitchB(config)#interface vlan 10
SwitchB(config-if)#ip address 10.1.10.3 255.255.255.0
SwitchB(config-if)#standby 10 ip 10.1.10.1
SwitchB(config-if)#standby 10 priority 90
SwitchB(config-if)#standby 10 preempt

SwitchB(config)#interface vlan 20
SwitchB(config-if)#ip address 10.1.20.3 255.255.255.0
SwitchB(config-if)#standby 20 ip 10.1.20.1
SwitchB(config-if)#standby 20 priority 110
SwitchB(config-if)#standby 20 preempt

```

```

R1#show standby
FastEthernet0/0.10 - Group 10
Local state is Active, priority 150, may preempt
Hello time 3 hold time 10
Next hello sent in 00:00:01.028
Hot standby IP address is 192.168.10.1 configured
Active router is local
Standby router is 192.168.10.3 expires in 00:00:08
Tracking interface states for 1 interface, 1 up:
Up Serial0/0
.....

```

```

Switch# show standby brief
          P indicates configured to preempt.
          |
Interface  Grp   Pri  P State      Active           Standby        Virtual IP
V11        10    150  P Active     local            192.168.10.253  192.168.10.250
V12        20    100  Standby    192.168.20.253  local          192.168.20.250
V13        30    150  P Active     local            192.168.30.253  192.168.30.250
V14        40    100  Standby    192.168.40.253  local          192.168.40.250
Switch#

```

Week 7 ACLs, NAT

ACL Acces Control List

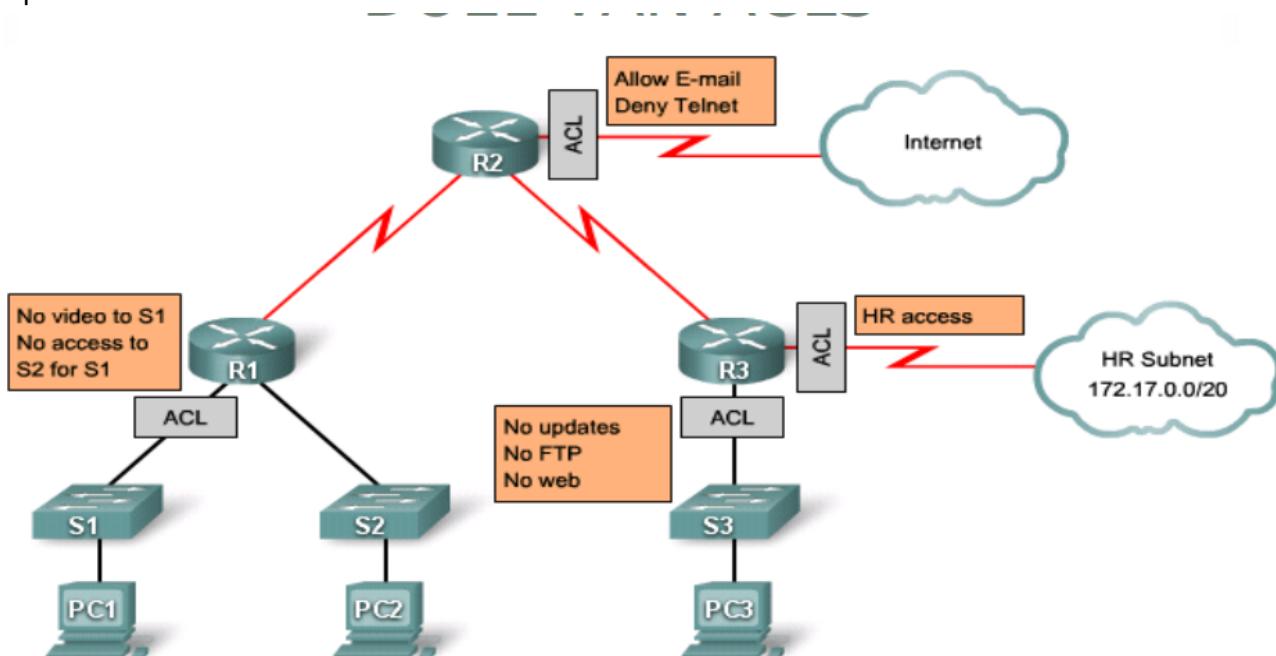
Een ACL is een lijst met *permit* en *deny* statements

```
| R2 (config)#access-list 1 permit 192.168.0.0 0.0.0.3  
| R2 (config)#access-list 1 permit 192.168.1.0 0.0.0.255
```

Verification:

```
| R2#sho access-lists  
Standard IP access list 1  
    10 permit 192.168.0.0 0.0.0.3  
    20 permit 192.168.1.0 0.0.0.255
```

Het doel hiervan is om verkeertypes te selecteren die je wilt filteren, analyseren, forwarden, of op een andere manier wilt verwerken.



Type ACLs

1. **Standaard ACL**: Filtert alleen IP packets gebaseerd op het source address

```
access-list 10 permit 192.168.30.0 0.0.0.255
```

2. **Extended ACL**: Filtert IP packets op meerdere attributen, zoals:

- Source en destination IP adres
- Source en destination TCP en UDP poorten

- Protocol typen (IP, ICMP, enz.)

```
access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
```

Configuratie

Nummering ranges:

Standard ACL: 1 - 99

Extended ACL: 100 - 199

```
R2(config)#access-list ?
<1-99>      IP standard access list
<100-199>    IP extended access list
```

standard
R2(config)#access-list 99 deny ? A.B.C.D Address to match any Any source host host A single host address ,

extended
R2(config)#access-list 100 deny ? ahp Authentication Header Protocol eigrp Cisco's EIGRP routing protocol esp Encapsulation Security Payload gre Cisco's GRE tunneling icmp Internet Control Message Protocol ip Any Internet Protocol ospf OSPF routing protocol tcp Transmission Control Protocol udp User Datagram Protocol ,

#Handig voor opdetoets

Standard ACL 1-99

Extended ACL 100-199

ACL benaming:

- namen kunnen nummers bevatten
- moet beginnen met een letter
- liefst in HOOFDLETTERS (ter onderscheid van een commando)
- geen spaties noch interpunctie

Commando nu iets anders:

```
R2(config)#ip access-list ?  
extended Extended Access List  
standard Standard Access List
```

```
R2(config)#ip access-list standard ?  
<1-99> Standard IP access-list number  
WORD Access-list name
```

Kan dus beide hier (nummer en naam) 

```
R2(config)#ip access-list standard MYLIST  
R2(config-std-nacl)#?  
<1-2147483647> Sequence Number  
default Set a command to its defaults  
deny Specify packets to reject  
exit Exit from access-list configuration mode  
no Negate a command or set its defaults  
permit Specify packets to forward  
remark Access list entry comment
```

Methode 1

```
R2(config)#access-list 1 permit 192.168.1.0 0.0.0.255  
R2(config)#access-list 1 permit 192.168.2.0 0.0.0.255  
R2(config)#
```

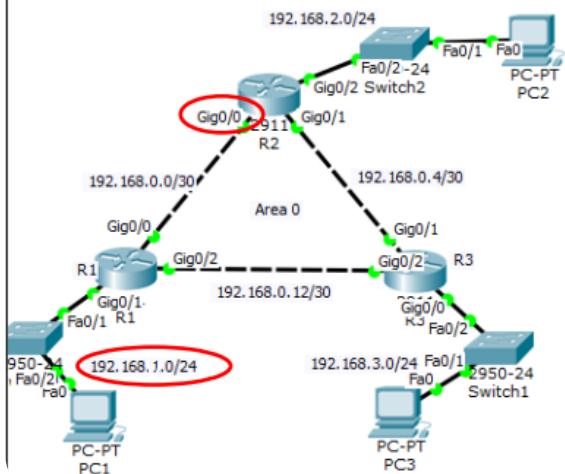
Methode 2

```
R2(config)#ip access-list standard MYLIST  
R2(config-std-nacl)#permit 192.168.1.0 0.0.0.255  
R2(config-std-nacl)#permit 192.168.2.0 0.0.0.255  
R2(config-std-nacl)#end  
R2#
```

Let op
verschil:
sequence
numbers
bij
"MYLIST"

```
%SYS-5-CONFIG_I: Configured from console by console  
R2#sho access-lists  
Standard IP access list MYLIST  
    10 permit 192.168.1.0 0.0.0.255  
    20 permit 192.168.2.0 0.0.0.255  
Standard IP access list 1  
    permit 192.168.1.0 0.0.0.255  
    , permit 192.168.2.0 0.0.0.255
```

Vb: Blok verkeer naar R2 van 192.168.1.0/24 via interface Gig0/0



Config:

```
R2(config)#access-list 99 deny 192.168.1.0 0.0.0.255
R2(config)#access-list 99 permit any
R2(config)#int gi0/0
R2(config-if)#ip access-group 99 in
```

Vóór toevoegen ACL:

```
R2#sho ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  192.168.0.0/24 is variably subnetted, 5 subnets, 2 masks
C    192.168.0.0/30 is directly connected, GigabitEthernet0/0
L    192.168.0.2/32 is directly connected, GigabitEthernet0/0
C    192.168.0.4/30 is directly connected, GigabitEthernet0/1
L    192.168.0.5/32 is directly connected, GigabitEthernet0/1
O    192.168.0.12/30 [110/2] via 192.168.0.1, 00:00:12, GigabitEthernet0/0
     [110/2] via 192.168.0.6, 00:00:12, GigabitEthernet0/1
O    192.168.1.0/24 [110/2] via 192.168.0.1, 00:00:12, GigabitEthernet0/0
     192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.2.0/32 is directly connected, GigabitEthernet0/2
L    192.168.2.1/32 is directly connected, GigabitEthernet0/2
O    192.168.3.0/24 [110/2] via 192.168.0.6, 00:00:12, GigabitEthernet0/1
```

Deze wilden we dus blokkeren

Na toevoegen ACL 99 op Gig0/0:

```
R2#sho ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  192.168.0.0/24 is variably subnetted, 5 subnets, 2 masks
C    192.168.0.0/30 is directly connected, GigabitEthernet0/0
L    192.168.0.2/32 is directly connected, GigabitEthernet0/0
C    192.168.0.4/30 is directly connected, GigabitEthernet0/1
L    192.168.0.5/32 is directly connected, GigabitEthernet0/1
O    192.168.0.12/30 [110/2] via 192.168.0.6, 00:13:55, GigabitEthernet0/1
     [110/2] via 192.168.0.6, 00:13:55, GigabitEthernet0/1
O    192.168.1.0/24 [110/3] via 192.168.0.6, 00:13:55, GigabitEthernet0/1
     192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.2.0/32 is directly connected, GigabitEthernet0/2
L    192.168.2.1/32 is directly connected, GigabitEthernet0/2
O    192.168.3.0/24 [110/2] via 192.168.0.6, 00:15:40, GigabitEthernet0/1
```

Gelukt, verkeer neemt route via Gig0/1

Inbound of Outbound

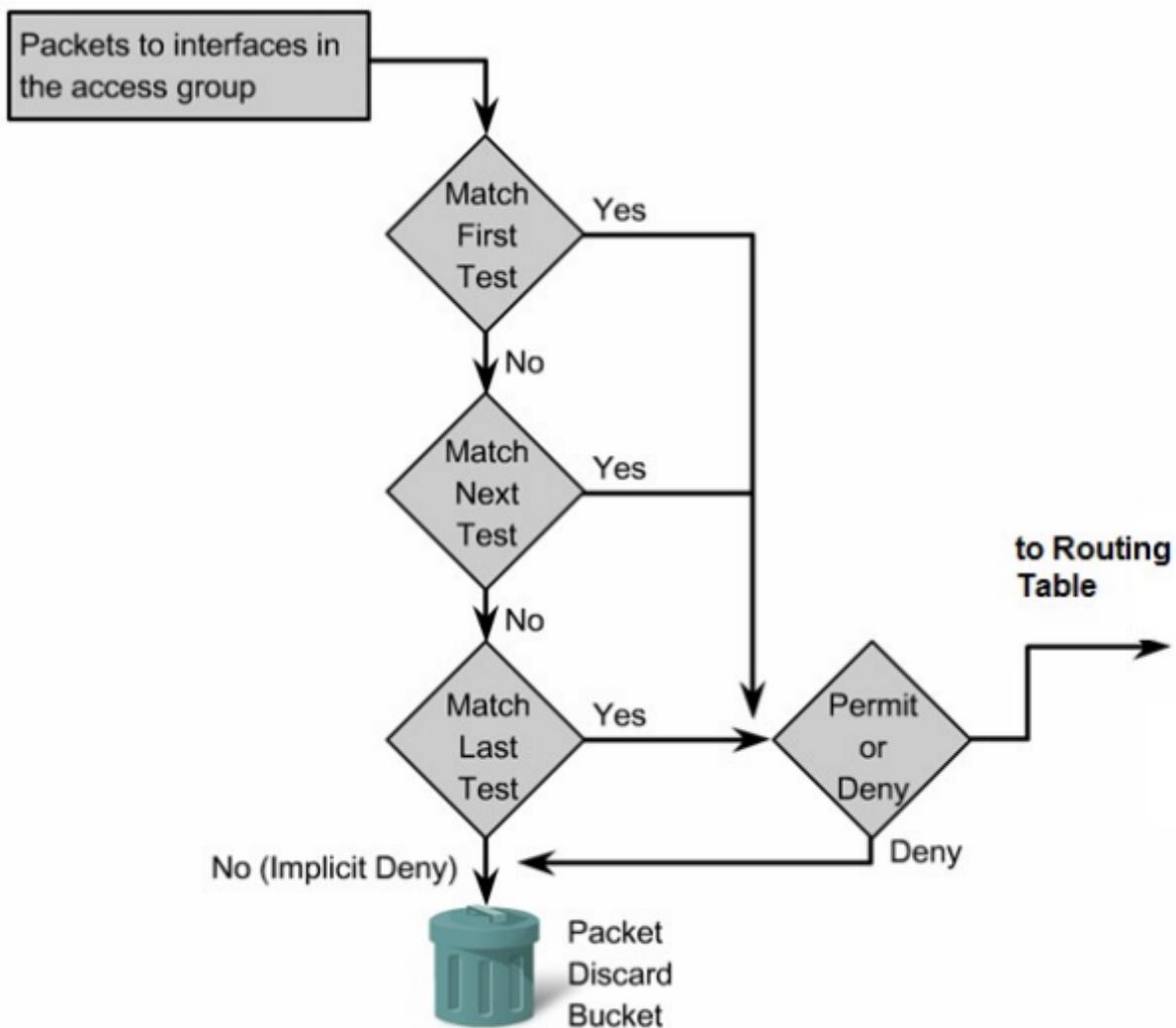
Inbound of outbound wordt per interface bepaald.

Hierbij is Inbound [in] het ingaand verkeer

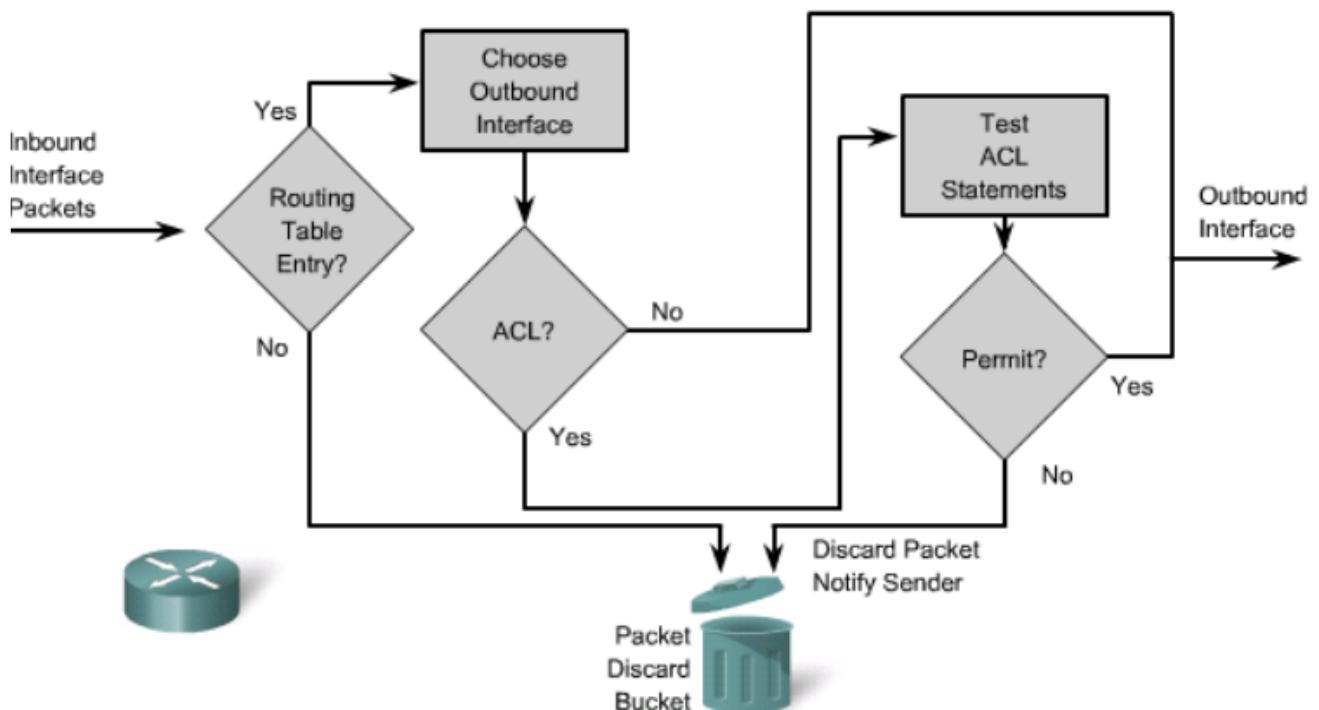
En

Outbound [out] is het uitgaand verkeer

Inbound



Outbound



Standard of Extended

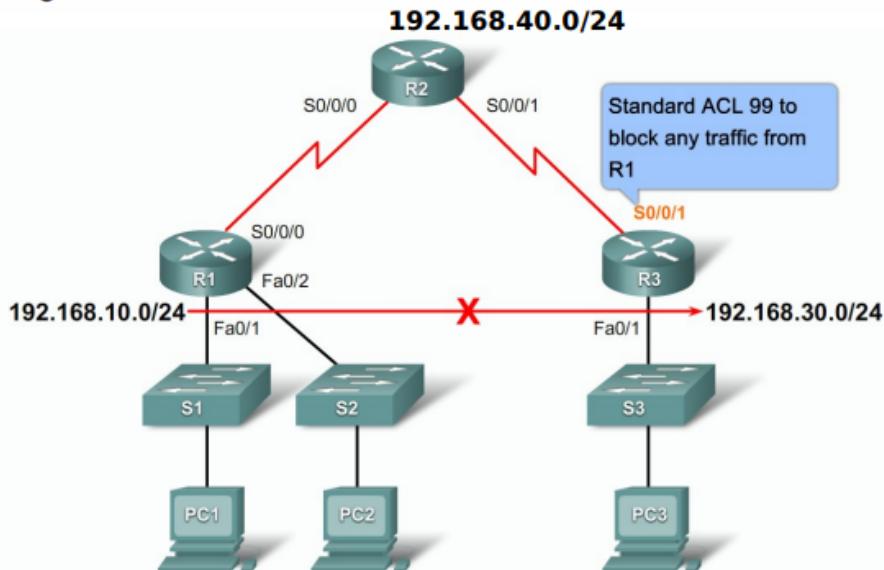
Standaard

Standard is dichtbij de **destination**

Als je dit niet doet dan zal een **deny** statement ervoor zorgen dat heel het source netwerk geen toegang heeft tot een enkel ander netwerk.

Dat is omdat een standaard ACL alleen kan filteren **vanaf** een bepaald netwerk en niet **naar** een bepaald netwerk. Er kan ook geen onderscheid gemaakt worden tussen protocols.

vb: LAN 192.168.30.0/.24 mag verkeer van netwerk 192.168.10.0/24 niet ontvangen, gebruik een standard ACL.

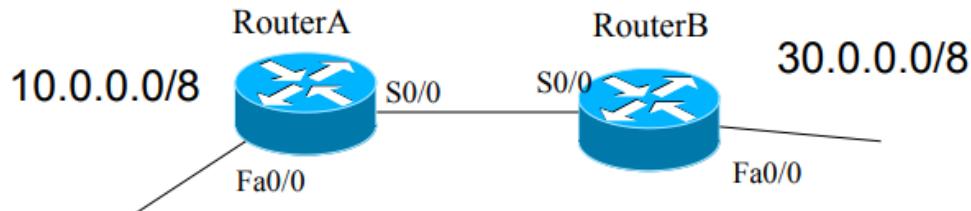


De standard ACL wordt geplaatst "near the destination", hier op de inbound interface van de ontvangende router (R3):

```
R3(config)#access-list 99 deny 192.168.10.0  0.0.0.255
R3(config)#access-list 99 permit any
R3(config)#int S0/0/1
R3(config)#ip access-group 99 in
```

Doel: blokkeren van verkeer van netwerk 10.0.0.0/8 naar netwerk 30.0.0.0/8

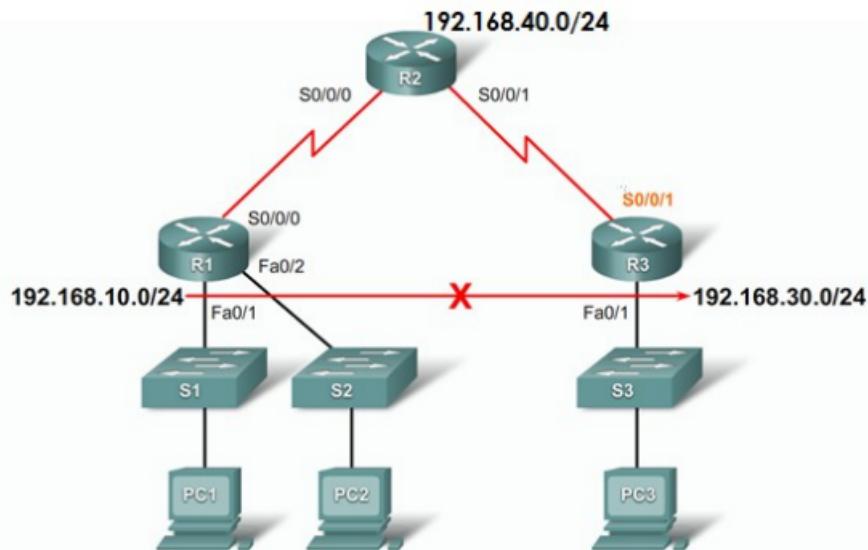
Configureer een standard ACL op routerB



```
RouterB(config)#access-list 10 deny 10.0.0.0  0.255.255.255
RouterB(config)#access-list 10 permit any
RouterB(config)#int fastethernet 0/0
RouterB(config-if)#ip access-group 10 out
```

Hieronder zie je wat er gebeurd als de standard ACL vlakbij de source is

LAN 192.168.30.0/24 mag verkeer van netwerk 192.168.10.0/24 niet ontvangen.



De standard ACL wordt geplaatst "near the source", bijv. op S0/0/0 van R1

```
R1 (config) #access-list 99 deny 192.168.10.0  0.0.0.255
R1 (config) #access-list 99 permit any
R1 (config) #int S0/0/0
R1 (config) #ip access-group 99 out
```

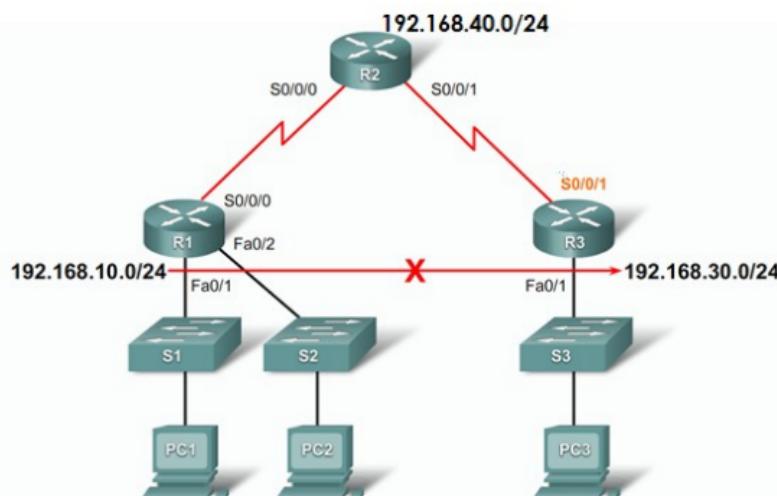
Extended

Extended is dichtbij de source

Als een extended ACL dichtbij een destination wordt geplaatst dan wordt de bandbreedte verspild. Dit is omdat de packets pas gedropt worden als ze bij de destination komen (dit geldt ook voor standaard ACL).

Een extended ACL wordt daarom zo dicht mogelijk bij de source geplaatst. Dit kan omdat deze veel specifieker is dan een standaard ACL

LAN 192.168.30.0/24 mag verkeer van netwerk 192.168.10.0/24 niet ontvangen.



De extended ACL wordt geplaatst "near the source", dus op Fa0/1 van R1

```
R1(config)#access-list 101 deny ip 192.168.10.0 0.0.0.255 192.168.30.0 0.0.0.255
R1(config)#access-list 101 permit ip any any
R1(config)#int Fa0/1
R1(config-if)#ip access-group 101 in
```

Syntax voor een extended ACL: access-list [nr] [permit|deny] [ip|icmp|tcp|udp] [source adres] [mask] [destination adres] [mask] [eq|ls|gt] [portnr|name] [log]

- access-list [nr] = het unieke nummer van de ACL
- [permit|deny] = het verkeer wordt toegelaten of geweigerd
- [ip|icmp|tcp|udp] = het netwerk protocol of transport protocol dat gefilterd moet worden
- [source adres] = het source IP/netwerk adres dat gefilterd moet worden
- [mask] = wildcard mask dat bepaalt welk gedeelte van het source adresveld gefilterd moet worden
- [port number] = poortnummer van belang (alleen als layer-4 protocol is geselecteerd)

Als het netwerkverkeer specieker moet worden gefilterd, kunnen de volgende opties worden gebruikt:

[eq|ls|gt]

- eq = gelijk aangepast
- ls = kleiner dan
- gt = groter dan

[portnr|name] = het poortnummer van het upper layer protocol (HTTP, FTP) dat gefilterd moet worden

[log] deze optie logt al het verkeer dat niet door de ACL wordt toegelaten (dit heet een access-list violation)



In dit voorbeeld mag node 10.0.0.2 geen toegang krijgen tot netwerk 30.0.0.0/8, alle andere nodes in het netwerk 10.0.0.0/8 krijgen deze beperking niet

```

RouterA(config)#access-list 100 deny ip host 10.0.0.2 30.0.0.0 0.255.255.255
RouterA(config)#access-list 100 permit ip any any
RouterA(config)#interface fastethernet 0/0
RouterA(config-if)#ip access-group 100 in

```

Wanneer we al het IP verkeer willen filteren ongeacht het transport protocol en upper layer protocol, definiëren we het netwerkprotocol als 'ip'

Wanneer er een bepaald protocol moet worden gefilterd, kan ook moet het bijbehorende transport layer protocol gedefinieerd worden:

```

RA(config)#access-list 100 permit udp host 10.0.0.2 30.0.0.0
0.255.255.255

```

'ICMP' wordt gebruikt om management protocollen te filteren zoals: ping en traceroute:

```

RA(config)#access-list 100 permit icmp host 10.0.0.2 30.0.0.0
0.255.255.255

```

De drie Per's

Een ACL moet aan de volgende eisen voldoen:

Een ACL...

1. Per protocol
2. Per interface
3. Per richting (in/uit)

NAT (Network Address Translation)

NAT

1. **Static NAT** 1 op 1 mapping tussen lokale en globale adressen.

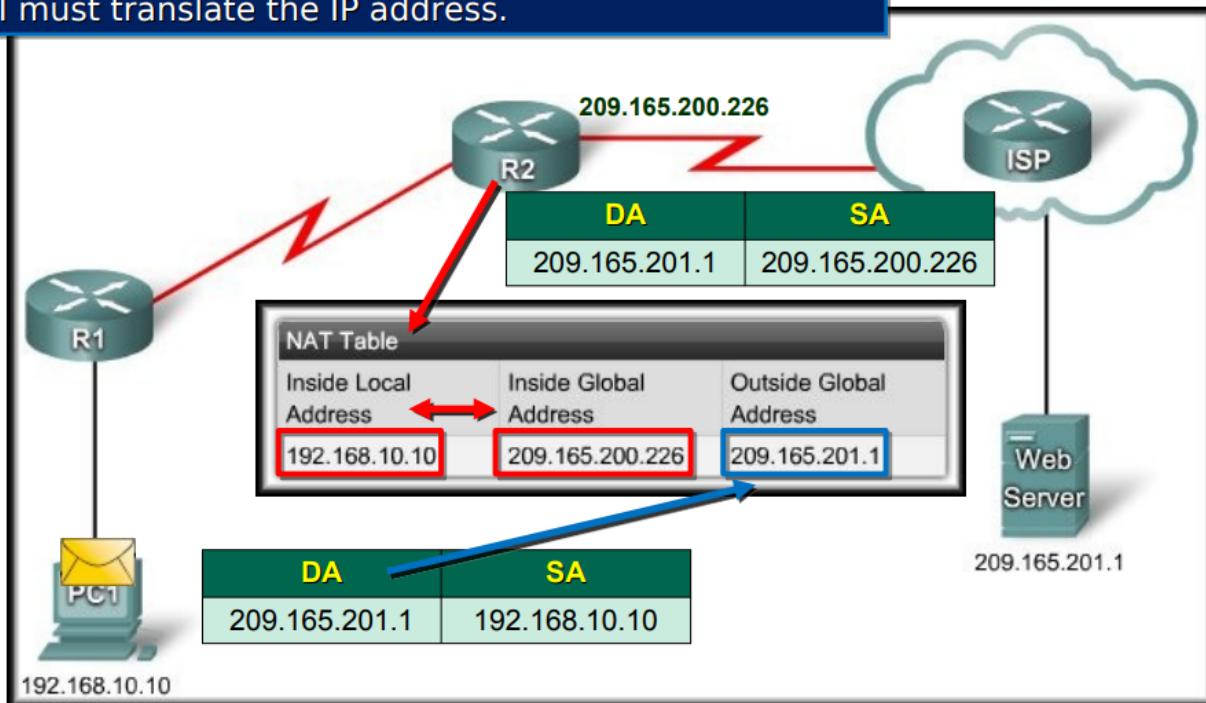
2. **Dynamic NAT** een groot aantal private IP adressen mappen op een klein aantal publieke adressen.
 3. **PAT (Port Address Translation; aka *Dynamic NAT with overload*)** Maakt gebruik van het poortnummer om een groot aantal private IP adressen te mappen op een klein aantal publieke adressen (vaakst gebruikt).
- Full cone NAT (=normaal PAT)
 - Elk device kan van buiten een verbinding maken als het de destination poort kent. Kan statisch toegewezen zijn. NAT kan de port mappen naar een andere destination port, dit heet port forwarding.
 - Address Restricted Cone NAT
 - De NAT router houdt de source port en de destination adres bij nadat verkeer naar buiten is verstuurd. Dus inkomend verkeer moet die poort als destination poort gebruiken en ook het source IP-adres moet gelijk zijn aan het opgeslagen destination adres.
 - Port Restricted Cone NAT
 - Net als hierboven maar nu moet ook de source port van het inkomende segment gelijk aan die van opgeslagen destination port.
 - Symmetrisch NAT
 - Creëert unique mappings. Elke source port wordt vertaald naar een random (ephemeral) port. Voor elke verandering in het destination IP-adres wordt zo'n mapping gedaan. Dit levert een extra beveiligingslaag.

Terminologie

- Inside Local Addresses - Adres van eigen host gezien vanuit het eigen netwerk.
- Inside Global Addresses- Adres van eigen host gezien door de provider.
- Outside Local Addresses- Adres van een host in een ander bedrijfsnetwerk.
- Outside Global Addresses- Adres van een andere host, gezien vanaf de buitenwereld (b.v. een ge-NAT IP-adres of een remote server met publiek adres)

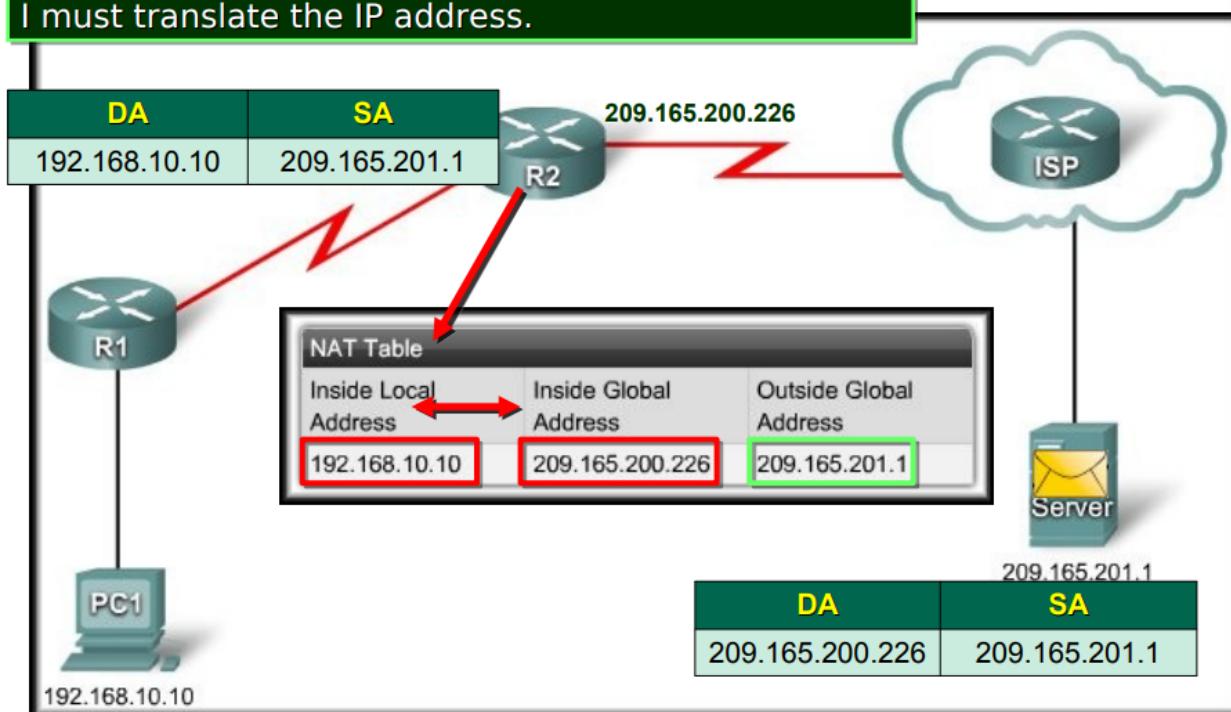
R2: I have a packet for the outside network.
I must translate the IP address.

Send

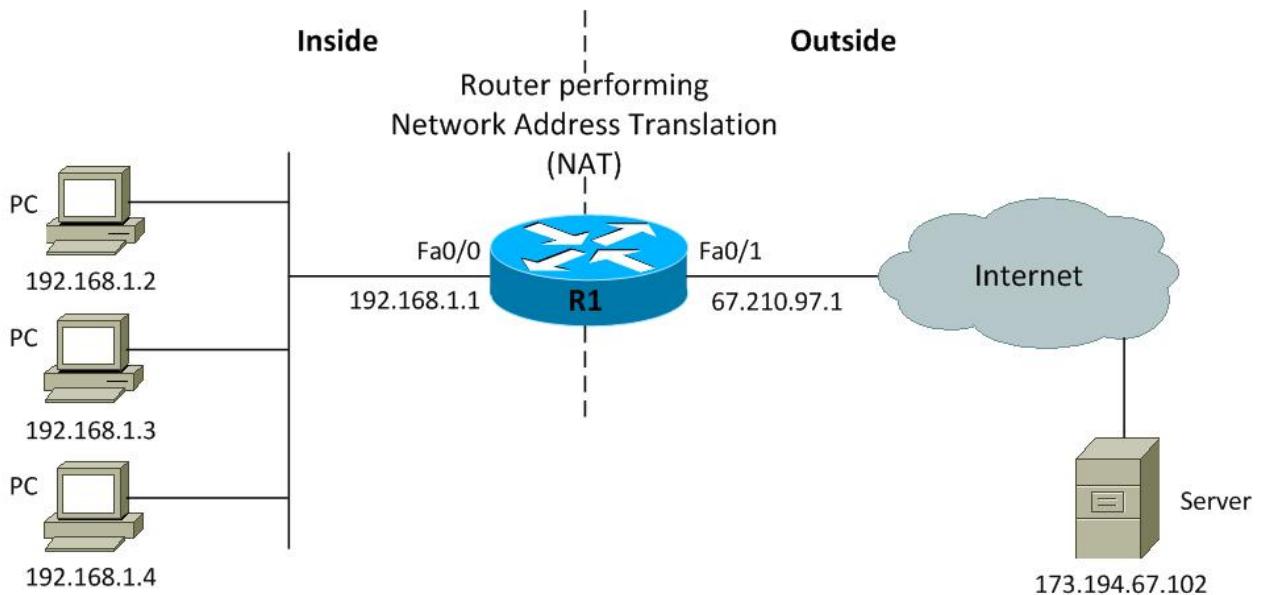


R2: I have a packet for the inside network.
I must translate the IP address.

Receive



Nat overload (PAT)



NAT Translation Table

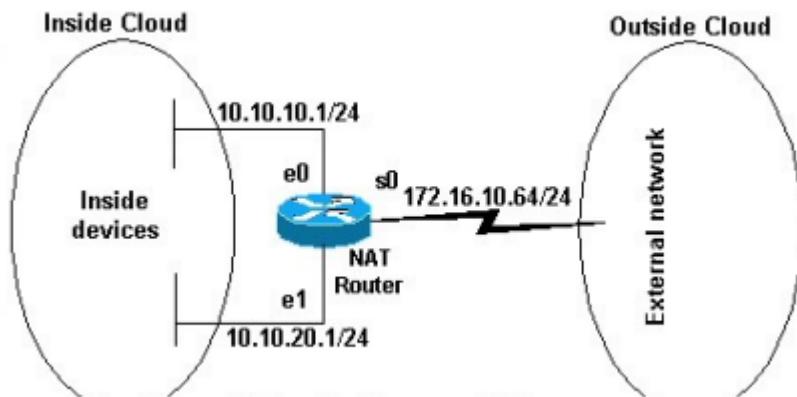
Protocol	Inside Local IP : Port	Inside Global IP : Port
ICMP	192.168.1.2 : 18	67.210.97.1 : 18
ICMP	192.168.1.3 : 19	67.210.97.1 : 19
ICMP	192.168.1.4 : 20	67.210.97.1 : 20

PAT

- Gebruikt 16 bit poortnummers
- De PAT router voegt de poortnummers toe aan de inside global adressen, welke opgeslagen worden in de NAT tabel. In principe wordt het source portnummer gebruikt in het segment gebruikt, als dat overlapt met een bestaand adres wordt een alternatief poortnummer gezocht worden.
- Als de server een packet terugstuurt, zal nu het destination poortnummer uit het packet gehaald worden en vergeleken worden met de entries in de NAT table. Zo kan het lokale ip adres teruggevonden worden
- Op deze manier wordt ook extra beveiliging toegevoegd omdat er gecheckt kan worden of de packets van die remote server inderdaad gewenst waren.

Configuratie

Stel we hebben dit netwerk



Dynamic NAT

Define interface Ethernet0 with an IP address and as a NAT inside interface (do the same for interface Ethernet1):

```
interface ethernet 0
ip address 10.10.10.1 255.255.255.0
ip nat inside
```

Identify traffic using an ACL,

```
access-list 7 permit 10.10.10.0 0.0.0.255
access-list 7 permit 10.10.20.0 0.0.0.255
```

Define interface Serial 0 as the NAT outside interface:

```
interface serial 0
ip address 172.16.10.64 255.255.255.0
ip nat outside
```

Define a NAT pool with a range of addresses to use (172.16.10.1 - 172.16.10.63):

```
ip nat pool NO_OVERLOAD_POOL 172.16.10.1 172.16.10.63 prefix 24
```

Packets received on the inside interfaces are permitted by ACL 7 and translated by the NAT pool.
ip nat inside source list 7 pool NO_OVERLOAD_POOL

Dynamic NAT met overload

Define interface Ethernet0 with an IP address and as a NAT inside interface (do the same for interface Ethernet1):

```
interface ethernet 0
ip address 10.10.10.1 255.255.255.0
ip nat inside
```

Identify traffic using an ACL,

```
access-list 7 permit 10.10.10.0 0.0.0.255
access-list 7 permit 10.10.20.0 0.0.0.255
```

Define interface Serial 0 as the NAT outside interface:

```
interface serial 0
ip address 172.16.10.64 255.255.255.0
ip nat outside
```

Define a NAT pool with a range of the single address 172.16.10.1

```
ip nat pool OVERLOAD_POOL 172.16.10.1 172.16.10.1 prefix 24
```

Packets received on the inside interfaces are permitted by ACL 7 and translated by the NAT pool to use.

```
ip nat inside source list 7 pool OVERLOAD_POOL overload
```

Port forwarding

