VLANs

VLAN + VLAN ACCES

Creating Vlan

S(config)# vlan 10
S(config-vlan)# name example

Ports - VLAN - Switch

S(config) interface [range]
S(config-if)# switchport mode access
S(config-if)# switchport access vlan 10

Management VLAN

S(config)# interface VLAN 99
S(config-if)# ip address A.B.C.D a.b.c.d

SHOW/VERIFY

S#show vlan [brief| id 10 |name exmaple |summary]
S# show interfaces fa 0/1
Switch#show interfaces vlan 99 switchport

Change/delete

S(config)# no vlan 20
S(config-if)#no switchport access vlan 10

S(config-vlan)#
S#delete flash:vlan.dat
S#erase startup-config
S#reload

VLAN Trunk

Creating Trunk(s)

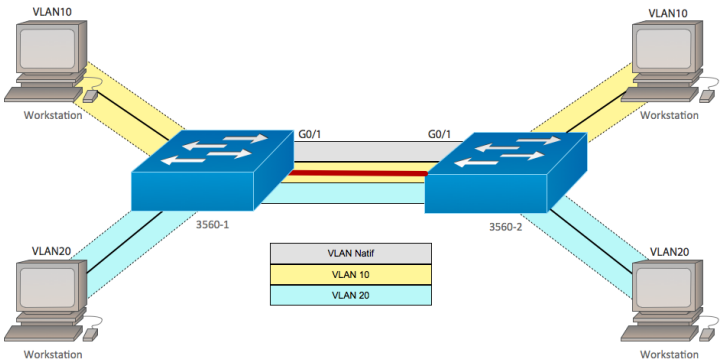
S(config-if)#
-> switchport mode trunk
-> switchport trunk native vlan 200
-> switchport trunk allowed vlan 10,20

SHOW/VERIFY

S#show interfaces trunk
S#show interfaces g0/1 switchport

Change/delete

S(config-if)# switchport trunk allowed add 99
S(config-if)#no switchport trunk

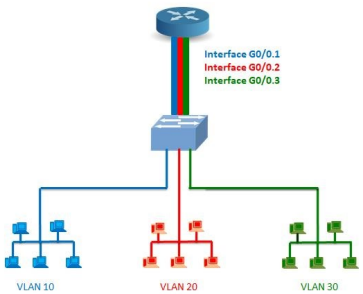


Inter-Vlan Routing Router on a stick

Router on a stick

R(config)#interface G0/0 .10
R(config-if)# encapsulation dot1q 10
R(config-if)#Ip address A.B.C.D a.b.c.d

R(config)#interface G0/0 .200
R(config-if)# encapsulation dot1q 200 native



Layer 3 switch

SVI VLAN interfaces

S(config)#Interface VLAN ID
S(config-if)# Description
S(config-if)# Ip add A.B.C.D A.B.C.D
S(config-if)#

Routing port

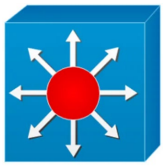
S(config-if)# no switchport
S(config-if)# ip add A.B.C.D A.B.C.D
S(config-if)#

Enable IP routing

S(config)# ip routing

SHOW/VERIFY

show ip interfaces brief
show ip route
show vlan



STP & Etherchannel

Spanning Tree Configuration

Spannig Tree configuration (PVST+)

```
S(config)# spanning-tree VLAN xx priority 4096
  // Root bridge =  lowest prio -> prio= multi.4096

S(config)# spanning-tree VLAN xx root primary
  //Root bridge dynamic (automatic prio)
S(config)# spanning-tree VLAN xx root secondary
 //backup Root bridge dynamic (automatic prio)
```

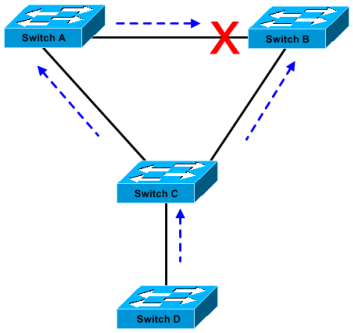
Rapid PVST+

```
S(config)# spanning-tree mode rapid-pvst

S(config-if)# spanning-tree link-type point-to-point
```

SHOW/VERIFY

```
S# show spanning-tree
S# show spanning-tree active
S# show spanning-tree vlan xx
```



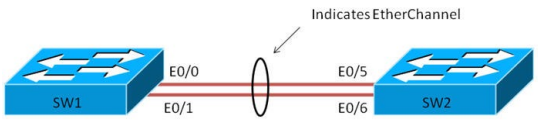
EtherChannel

EtherChannel

```
S(config-if-range)# shutdown
S(config-if-range)# duplex auto
S(config-if-range)# speed 100
S(config-if-range)# channel-group 1 mode active
S(config-if-range)# no shutdown
```

SHOW/VERIFY

```
S# show interfaces port-channel
S# show etherchannel summary
S# show etherchannel port-channel
S# show interface f0/1 etherchannel
```



DHCPv4, DHCPv6 & SLAAC

DHCPv4 pools

Step 1: Exclude IPv4 addresses

R(config)# ip dhcp excluded-address low-address [high-address]

Step 2: Define and configure a DHCPv4 pool

R(config)# ip dhcp pool *pool-name*
R(dhcp-config)# network network-number [mask | / prefix-length]
R(dhcp-config)# default-router address [A.B.C.D]
R(dhcp-config)# dns-server [A.B.C.D]
R(dhcp-config)# domain-name Example.com
R(dhcp-config)# lease {days [hours [minutes]] | infinite}
R(dhcp-config)# netbios-name-server address [address]

SHOW/VERIFY/delete

R#show running | section DHCP
R1# show ip dhcp binding
R1# show ip dhcp server statistics
R(config)#no dhcp …
R(config)#debug IP DHCP server events

DHCPv4 service & relay

DHCPv4 Relay command *

R(config-if)#ip helper-address A.B.C.D // *broadcasts to the helper address (DNS server)*

DHCPv4 Router as a client

R(config-if)# ip address dhcp – no shut

Disable

R(config) no service dhcp

**services forwared by default*
Port 37: Time
Port 49: TACACS
Port 53: DNS
Port 67: DHCP/BOOTP server
Port 68: DHCP/BOOTP client
Port 69: TFTP
Port 137: NetBIOS name service
Port 138: NetBIOS datagram service

SHOW/VERIFY

C: ipconfig \release

R#show ip interface g0/0
R# show interface g0/1
R# show ip dhcp conflict
R# show interfaces
R# show run config | section interface g0/0
;remark 'service dhcp' = default = not shown
R# show ip interface ; check the relay address
R# show run config | include no service dhcp

DHCPv6

Enable IPv6 routing

R(cconfig)#ipv6 unicast-routing

DHCPv6 pools

R(config)#ipv6 dhcp pool [name]
R(config-dhcpv6)#address prefix X:X:X:X::X/<0-128> lifetime infinite
R(config-dhcpv6)#dns-server X:X:X:X::X
R(config-dhcpv6)#domain-name [name]

DHCPv6 interface

R(config-if)# ipv6 address X:X:X:X::X/<0-128>
R(config-if)# ipv6 dhcp server [pool name]
R(config-if)# ipv6 nd other-config-flag

DHCPv6 client

R(config)# interface g0/0
R(config-if)#ipv6 enable
R(config-if)#ipv6 address dhcp

SHOW/VERIFY

R#show ipv6 dhcp pool
Router# show ipv6 dhcp binding
Router# show ipv6 interface g0/0
Router# debug ipv6 dhcp detail

Link-local IPv6 address - fe80::1
GUA and subnet - 2001:db8:acad:1::1
IPv6 all-nodes group - ff02::1

FHRP

HSRP

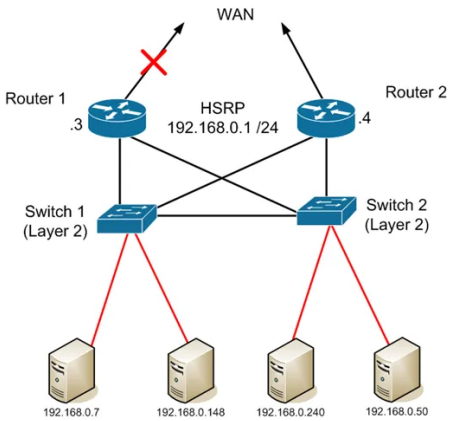
Hot Standby Routing Protocol

```
R1(config-if)# ip address A.B.C.D A.B.C.D
R1(config-if)# standby version 2
R1(config-if)# standby 1 ip A.B.C.D
  //  =virtual ip address => DG van het netwerk
R1(config-if)# standby 1 priority 110
R1(config-if)# standby 1 preempt
R1(config-if)# no shutdown
```

SHOW/VERIFY

```
Router# show standby
Router# show standby brief

Router# debug standby packets|terse|errors|events
```



Switch security

Port Security

Port Security

```
S(config-if)# switchport port-security  |?

S(config-if)# switchport port-security maximum value

S(config-if)# switchport port-security mac-address aaaa.bbbb.1234

S(config-if)# switchport port-security mac-address sticky

S(config-if)# switchport port-security aging { static | time time | type {absolute | inactivity}}
```

Port Security Violation Modes

```
Switch(config-if)# switchport port-security violation { protect | restrict | shutdown}
```

// good practice to disable unused ports → S1(config-if-range)# shutdown
// re-enable the port ⇒ shutdown -> no shutdown

show/verify

```
S# show port-security
S# show port-security interface fa0/1
S# show run interface fa0/1 //verify sticking MAC addresses
S# show port-security address
```

VLAN hopping

Mitigate VLAN hopping

```
S(config)# interface range fa0/x-x
S(config-if-range)# switchport mode access
S(config-if-range)# switchport access vlan [id]

S(config)# interface range fa0/x-x
S(config-if-range)# switchport mode trunk
S(config-if-range)# switchport nonegotiate
S(config-if-range)# switchport trunk native vlan [id]
```

DHCP snooping

Mitigate DHCP attacks

```
S(config)# ip dhcp snooping  //DHCP snooping is enabled
S(config-if)# ip dhcp snooping trust // intf explicitly trusted
S(config-if-range)# ip dhcp snooping limit rate 6
S(config)# ip dhcp snooping vlan 5,10,50-52

S# show ip dhcp snooping
```

DAI

Mitigate ARP attacks

```
S(config)# ip dhcp snooping
S(config)# ip dhcp snooping vlan 10
S(config)# ip arp inspection vlan 10
S(config)# interface fa0/24
S(config-if)# ip dhcp snooping trust
S(config-if)# ip arp inspection trust
```

Portfast - BPDU

Mitigate STP attacks

```
S(config)# spanning-tree portfast
S(config)# spanning-tree portfast default
S(config)# spanning-tree portfast bpduguard default

S(config)# int f0/1
S(config-if)# spanning-tree portfast
S(config-if)# spanning-tree bpduguard enable
```

Routing

Static Routing

Config IPv4 Routes

```
R(config)#ip route network-address subnet-mask
{ip-address | exit-intf}
// routes met next-hop adres
R(config)#ip route A.B.C.D A.B.C.D A.B.C.D
// Routes met exit interface
RouterB(config)#ip route 172.16.1.0 255.255.255.0
Serial 0/0
//Routes fully specified
R(config)#ip route A.B.C.D A.B.C.D  INT A.B.C.D
```

Default Route

```
R(config)# ip route 0.0.0.0 0.0.0.0 { ip-address | exit-intf}
```

Floating Static Routes

```
R(config)# ip route A.B.C.D A.B.C.D A.B.C.D NR
```

SHOW/VERIFY

```
R#show ip route
R# ip route network-address subnet-mask {ip-address | exit-intf}
Rr# show ip route static
```

Change/delete

```
R(config)# no ip route network-address subnet-mask {ip-address | exit-intf}
```

Troubleshoot

```
Router# debug ip routing
Router# no debug ip routing (enkel debugging op routing
afzetten)
Router# no debug all
Tracert
// extended ping → R#ping 192.168.2.1 source g0/0
```

Dynamic Routing (RIP)

RIP

```
R(config)# router rip
R(config-router)# version 1 / 2
R(config-router)# no auto-summary
Router(config)# no ip classless
```

Add Routes

```
// Add a network
RouterB(config-router)# network 192.168.1.0

// default route
R(config)# ip route 0.0.0.0 0.0.0.0 INT
R(config-router)# default-information originate
```

SHOW/VERIFY

```
R# show ip protocols
R# show ip rip database
R#show ip interface brief
```

Change/delete

```
R# no ip route network-address subnet-mask {ip-address | exit-intf}
```

Troubleshoot

```
R# debug ip rip
```

Passive interface

```
R(config-router)# passive-interface INT
```

Exchange static routes

```
R(config-router)# redistribute static
```

OSPF

OSPF

```
R(config)# router OSPF 1
// Number is local process ID
R(config-router)# router-id A.B.C.D
```

Add Routes

```
// Add a network with wildcard mask
R(config-router)# network 192.168.1.0 0.0.0.255 area 0

// default route
R(config)# ip route 0.0.0.0 0.0.0.0 INT
R(config-router)# default-information originate
```

Add Routes on interface

```
R(config)# interface GigabitEthernet 0/0
R(config-if)# ip ospf 10 area 0
R(config-if)# interface GigabitEthernet 0/1
R(config-if)# ip ospf 10 area 0
```

Exchange static routes

```
R(config-router)# redistribute static
```

SHOW/VERIFY

```
R# show ip protocols
R# show ip ospf neighbors
R# show ip ospf topology
R# show ip route

R# clear ip ospf process
```

Passive interface

```
R(config-router)# passive-interface INT
```

Multi-Access: DR/BDR/DROther

```
R(config)# interface G0/0
R(config-if)# ip ospf priority 255
```

Modify

```
R(config)# interface serial 0/0
R(config-if)# no bandwidth 64
R(config-if)# ip ospf cost 1562

R(config-router)# auto-cost reference-bandwidth 10000
```

ACL

ALC

Numbered Standard ACL

```
R(config)# access-list access-list-number {deny|permit|remark} source [source-wildcard][log]

R(config)# access-list 10 remark Permit host from the 192.168.30.0 LAN
R(config)# access-list 10 permit 192.168.30.0 0.0.0.255

// Assign to interface
R(config)# interface g0/0
R(config-if)# ip access-group 10 in/out
```

Named Standard ACL

```
R(config)# ip access-list standard [name]
R(config-std-nacl)# deny 192.168.10.0 0.0.0.255
R(config-std-nacl)# permit any
R(config-std-nacl)# exit

// Assign to interface
R(config)# interface g0/0
R(config-if)# ip access-group [name] in/out
```

Named Extended ACL

```
R(config)# ip access-list extended [name]
R(config-ext-nacl)# permit tcp 192.168.10.10 0.0.0.255 any eq 80
R(config-ext-nacl)# permit tcp 192.168.10.10 0.0.0.255 any eq 443

// Assign to interface
R(config)# interface g0/0
R(config-if)# ip access-group [name] in/out
```

show/verify

```
R# show access-lists
```

NAT

NAT for IPv4

Static NAT

```
R(config)# ip nat inside source static local-ip global-ip
```

Port Forwarding

```
R(config)# ip nat inside source static protocol local-ip local-port global-ip global-port
```

Dynamic NAT

```
R(config)# ip nat pool name start-ip end-ip {netmask netmask | prefix-length prefix-length}
R(config)# access-list access-list-number permit source [source-wildcard]
R(config)# ip nat inside source list access-list -number pool name
```

NAT overload / PAT

```
R(config)# access-list access-list-number permit source [source-wildcard]
R(config)# ip nat inside source list access-list-number interface interface-name overload
```

Inside and Outside interface

```
R(config)# interface G0/0
R(config-if)# ip nat inside
R(config)# interface G0/1
R(config-if)# ip nat outside
```

show/verify

```
R# show ip nat translations
R# show ip nat statistics
R# clear ip nat statistics
R# debug ip nat
```