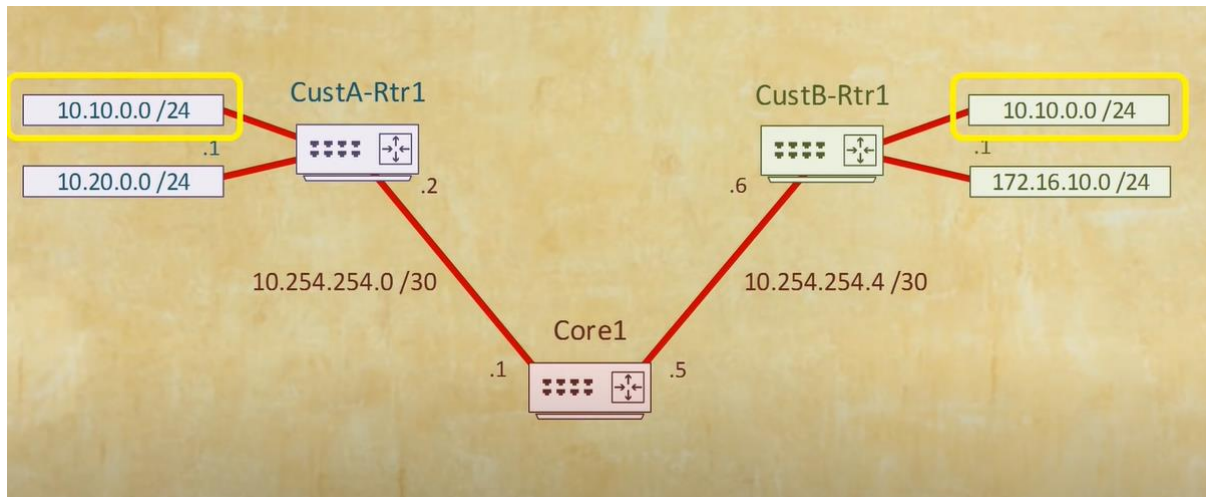


VRFs (Virtual Route Forwarding)

(All done on the Core Router as VRFs are locally significant)

(VRFs, or Virtual Routing and Forwarding, are virtual routing tables. They enable separation of one part of the network from another. There could be many reasons to do this. It could be for security, to separate the inside network from the DMZ. Or, it could be to separate BU's, or separate customers from each other)



Defining two empty VRFs:

```
Core1(config)#vrf definition CustA
Core1(config-vrf)#description Customer A
Core1(config-vrf)#address-family ipv4
Core1(config-vrf)#exit
Core1(config)#vrf definition CustB
Core1(config-vrf)#description Customer B
Core1(config-vrf)#address-family ipv4
Core1(config-vrf)#exit
```

Always check interface config before adding it to the VRFs (when you add an interface to a VRF all layer 3 config is lost):

```
Core1(config)#do show run int gi0/1
Building configuration...

Current configuration : 119 bytes
!
interface GigabitEthernet0/1
 ip address 10.254.254.1 255.255.255.252
 duplex auto
 speed auto
 media-type rj45
end
```

Configure IPs and test pings:

```
Core1(config)#int gi0/1
```

```
Core1(config-if)#ip address 10.254.254.1 255.255.255.252
```

```
Core1(config)#ping 10.254.254.2 (this will fail as it is using global routing table)
```

```
Core1(config)#ping vrf CustA 10.254.254.2
```

(Successful ping)

```
Core1(config)#int gi0/2
```

```
Core1(config-if)#ip address 10.254.254.5 255.255.255.252
```

```
Core1(config)#ping vrf CustB 10.254.254.6
```

(Successful ping)

Add Static Routes:

```
Core1(config)#ip route vrf CustA 10.10.0.0 255.255.255.0 10.254.254.2
```

```
Core1(config)#ip route vrf CustA 10.20.0.0 255.255.255.0 10.254.254.2
```

```
Core1(config)#ip route vrf CustB 10.10.0.0 255.255.255.0 10.254.254.6
```

```
Core1(config)#ip route vrf CustB 172.16.10.0 255.255.255.0 10.254.254.6
```

```
Core1#show ip route
```

(Global Routing Table will be completely empty as all the routes are under individual VRFs)

Core1#show ip route vrf CustA

```
Core1#show ip route vrf CustA

Routing Table: CustA
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
       i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

    10.0.0.0/8 is variably subnetted, 4 subnets, 3 masks
S       10.10.0.0/24 [1/0] via 10.254.254.2
S       10.20.0.0/24 [1/0] via 10.254.254.2
C       10.254.254.0/30 is directly connected, GigabitEthernet0/1
L       10.254.254.1/32 is directly connected, GigabitEthernet0/1
Core1#
```

Core1#show ip route vrf CustB

```
Core1#show ip route vrf CustB

Routing Table: CustB
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
       i - IS-IS, su - IS-IS summary, 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, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR

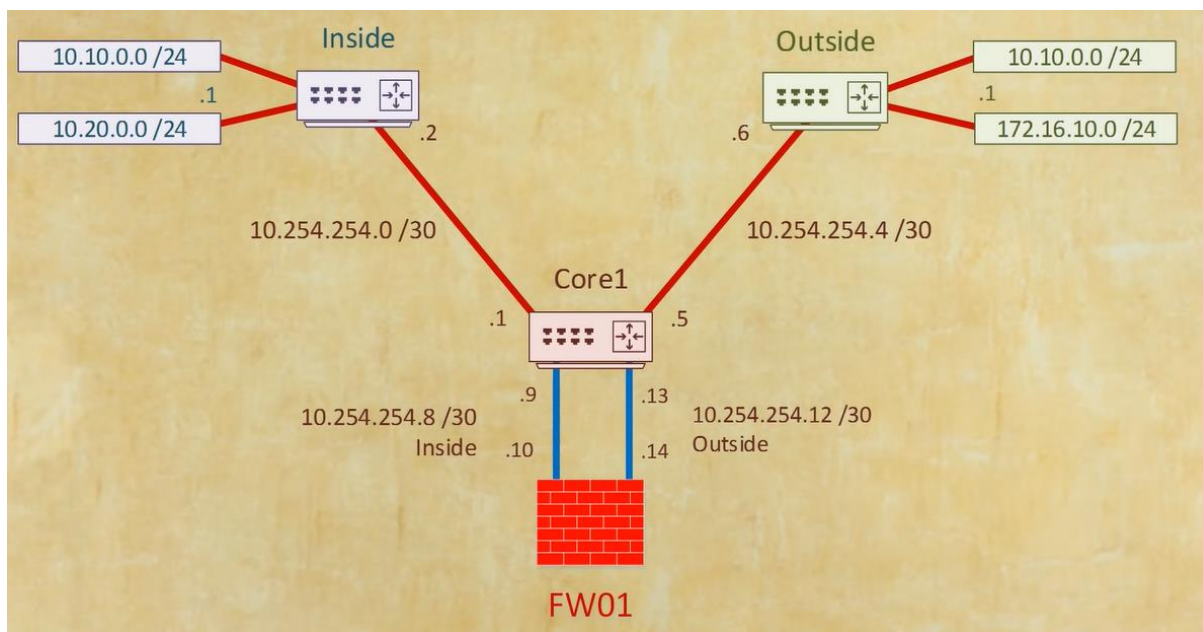
Gateway of last resort is not set

    10.0.0.0/8 is variably subnetted, 3 subnets, 3 masks
S       10.10.0.0/24 [1/0] via 10.254.254.6
C       10.254.254.4/30 is directly connected, GigabitEthernet0/2
L       10.254.254.5/32 is directly connected, GigabitEthernet0/2
       172.16.0.0/24 is subnetted, 1 subnets
S       172.16.10.0 [1/0] via 10.254.254.6
Core1#
```

```

Core1#ping vrf CustA 10.10.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.0.1, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
Core1#ping vrf CustA 10.20.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.20.0.1, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Core1#ping vrf CustB 10.10.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.0.1, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/4 ms
Core1#ping vrf CustB 172.16.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.1, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
Core1#
Core1#
Core1#
Core1#
Core1#
Core1#ping vrf CustB 10.20.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.20.0.1, timeout is 2 seconds:

```



(Firewall has already been configured)

```
Core1(config)#int gi 0/3
```

```
Core1(config-if)#desc Trunk to ASA
```

```
Core1(config-if)#no shut
```

```
Core1(config-if)#int gi 0/3.10
```

```
Core1(config-subif)#desc CustA Uplink
```

```
Core1(config-subif)#encapsulation dot1Q 10
```

```
Core1(config-subif)#vrf forwarding CustA
```

```
Core1(config-subif)#ip address 10.254.254.9 255.255.255.252
```

```
Core1(config-subif)#int gi 0/3.20
```

```
Core1(config-subif)#desc CustB Uplink
```

```
Core1(config-subif)#encapsulation dot1Q 20
```

```
Core1(config-subif)#vrf forwarding CustB
```

```
Core1(config-subif)#ip address 10.254.254.13 255.255.255.252
```

```
Core1#ping vrf CustA 10.254.254.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.254.254.10, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 2/2/3 ms
Core1#
Core1#ping vrf CustB 10.254.254.14
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.254.254.14, timeout is 2 seconds:
.!!!!
```

Now Configure Routing between two Customers (Avoid Overlapping Routes):

```
Core1(config)#ip route vrf CustA 10.254.254.4 255.255.255.252 10.254.254.10
Core1(config)#ip route vrf CustA 172.16.10.0 255.255.255.0 10.254.254.10
Core1(config)#
Core1(config)#ip route vrf CustB 10.20.0.0 255.255.255.0 10.254.254.14
Core1(config)#ip route vrf CustB 10.254.254.0 255.255.255.252 10.254.254.14
```

```
CustA-Rtr1#traceroute 172.16.10.1
Type escape sequence to abort.
Tracing the route to 172.16.10.1
VRF info: (vrf in name/id, vrf out name/id)
 1 10.254.254.1 2 msec 3 msec 5 msec
 2 10.254.254.10 3 msec * 3 msec
 3 10.254.254.13 14 msec 8 msec 6 msec
 4 10.254.254.6 8 msec * 8 msec
CustA-Rtr1#
```

```
CustB-Rtr1#trace
CustB-Rtr1#traceroute 10.20.0.1
Type escape sequence to abort.
Tracing the route to 10.20.0.1
VRF info: (vrf in name/id, vrf out name/id)
 1 10.254.254.5 4 msec 7 msec 2 msec
 2 10.254.254.14 6 msec * 4 msec
 3 10.254.254.9 7 msec 4 msec 5 msec
 4 10.254.254.2 7 msec
```