Segurança em Redes de Comunicações

High-Availability Firewalls Scenarios

VyOS

Active-Active Scenario

After the first boot of each firewall, load the default configuration and reboot:

```
sudo cp /opt/vyatta/etc/config.boot.default /config/config.boot
reboot
```

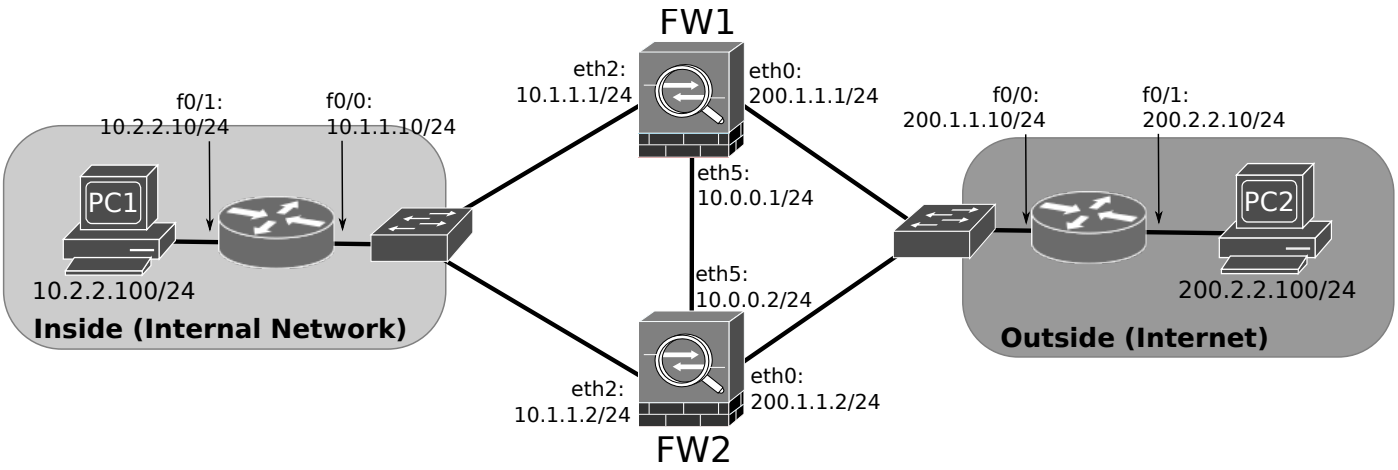
Check network interface names: `ip addr`

To change the keyboard layout: `set console keymap`

For QEMU GNS3 template use the following parameters: RAM: 512M, Console type: telnet (or none with auto start console checked), HDD Disk interface: ide, Network Adapters: 6, Network Name format: eth{0}.

For VirtualBox GNS3 template use the following parameters: RAM: 512M, Console type: telnet (or none with auto start console checked), Network Adapters: 6, Network Name format: eth{0}, check Network option “Allow GNS3 to use any ... adapter”.

VyOS user guide: https://docs.vyos.io/en/latest/



1. Configure the network depicted in the previous figure using GNS3 with PC1 and PC2 as VPCS, and the the firewalls as VyOS VM. Configure PCs addresses and gateways, and Routers addresses and static routes. For testing purposes configure asymmetric routing: Inside router routes via FW2, and Outside router routes via FW1. You must assume that the Internal Network as the 192.1.0.0/23 IPv4 public network.

```
RouterInside(config)# interface f0/0
RouterInside(config-if)# ip address 10.1.1.10 255.255.255.0
RouterInside(config-if)# no shutdown
RouterInside(config)# interface f0/1
RouterInside(config-if)# ip address 10.2.2.10 255.255.255.0
RouterInside(config-if)# no shutdown
RouterInside(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.2
–
RouterOutside(config)# interface f0/0
RouterOutside(config-if)# ip address 200.1.1.10 255.255.255.0
RouterOutside(config-if)# no shutdown
RouterOutside(config)# interface f0/1
RouterOutside(config-if)# ip address 200.2.2.10 255.255.255.0
RouterOutside(config-if)# no shutdown
RouterOutside(config)# ip route 192.1.0.0 255.255.254.0 200.1.1.1
```

For initial testing (before NAT configuration in FW1 and FW2) add also a static route to the private networks:

```
RouterOutside(config)# ip route 10.0.0.0 255.0.0.0 200.1.1.1
```

>> Verify the Routers routing tables with `sh ip route`

2. Configure the firewalls names, IPv4 addresses and basic static routing using the following commands:

- For FW1:

```
$ configure
# set system host-name FW1
# set interfaces ethernet eth0 address 200.1.1.1/24
#set interfaces ethernet eth2 address 10.1.1.1/24
# set interfaces ethernet eth5 address 10.0.0.1/24
# set protocols static route 0.0.0.0/0 next-hop 200.1.1.10
# set protocols static route 10.2.2.0/24 next-hop 10.1.1.10
# commit
# exit
```

- For FW2:

```
$ configure
# set system host-name FW2
# set interfaces ethernet eth0 address 200.1.1.2/24
#set interfaces ethernet eth2 address 10.1.1.2/24
# set interfaces ethernet eth5 address 10.0.0.2/24
# set protocols static route 0.0.0.0/0 next-hop 200.1.1.10
# set protocols static route 10.2.2.0/24 next-hop 10.1.1.10
# commit
# exit
```

>> Verify the configured addresses with: `$ show interfaces`

>> Verify the Firewalls' routing tables with: `$ show ip route`

>> Test the full connectivity between PC1 and PC2.

If working as expect save the configuration:

```
$ configure
# save
```

3. Configure the firewalls NAT/PAT mechanisms:

- For FW1 and FW2:

```
$ configure
# set nat source rule 10 outbound-interface eth0
# set nat source rule 10 source address 10.0.0.0/8
# set nat source rule 10 translation address 192.1.0.1-192.1.0.10
# commit
# exit
```

>> Test the full connectivity between PC1 and PC2 using UDP pings: `PC1> ping 200.2.2.100 -P 17 -p 5001`

>> Use the following command to verify the active NAT translations in both Firewalls: `$ show nat source translations`

>> Explain the lack of connectivity.

4. Configure the firewalls high-availability (Virtual Router Redundancy Protocol - VRRP) and connection state synchronization (conntrack-sync) mechanisms:

- VRRP in FW1 and FW2:

```
# set high-availability vrrp group FWCluster vrid 10
# set high-availability vrrp group FWCluster interface eth5
# set high-availability vrrp group FWCluster virtual-address 192.168.100.1/24
# set high-availability vrrp sync-group FWCluster member FWCluster
# set high-availability vrrp group FWCluster rfc3768-compatibility
```

- conntrack-sync in FW1 and FW2:

```
# set service conntrack-sync accept-protocol 'tcp,udp,icmp'
# set service conntrack-sync failover-mechanism vrrp sync-group FWCluster
# set service conntrack-sync interface eth5
# set service conntrack-sync mcast-group 225.0.0.50
# set service conntrack-sync disable-external-cache
```

>> Capture packets in the link between FW1 and FW2 and analyze/identify the exchanges packets.

>> Test the full connectivity between PC1 and PC2 using UDP pings: `PC1> ping 200.2.2.100 -P 17 -p 5001`

>> Use the following command to verify the active NAT translations in both Firewalls

```
$ show nat source translations
```

>> Use the following commands to analyze the current tracked and synced connections:

VyOS show commands:

```
$ show conntrack table ipv4
$ show conntrack-sync internal-cache
$ show conntrack-sync statistics
```

Native Linux services:

```
$ sudo conntrack -L
$ sudo conntrackd -i
$ sudo conntrackd -s
```

Note:VRRP stand for “Virtual Router Redundancy Protocol” which is not necessary in a Active-Active scenario, but VyOS OS imposes it as a requirement (assumes Active-Backup scenario).

Note2: The command “`set...disable-external-cache`” forces an immediate use of the states received from the other Firewall, enables the Active-Active scenario.

5. Configure a flow control rule in FW1 and FW2 to allow only UDP traffic (from Inside to Outside zones) that use UDP destination ports between 5000 and 6000.

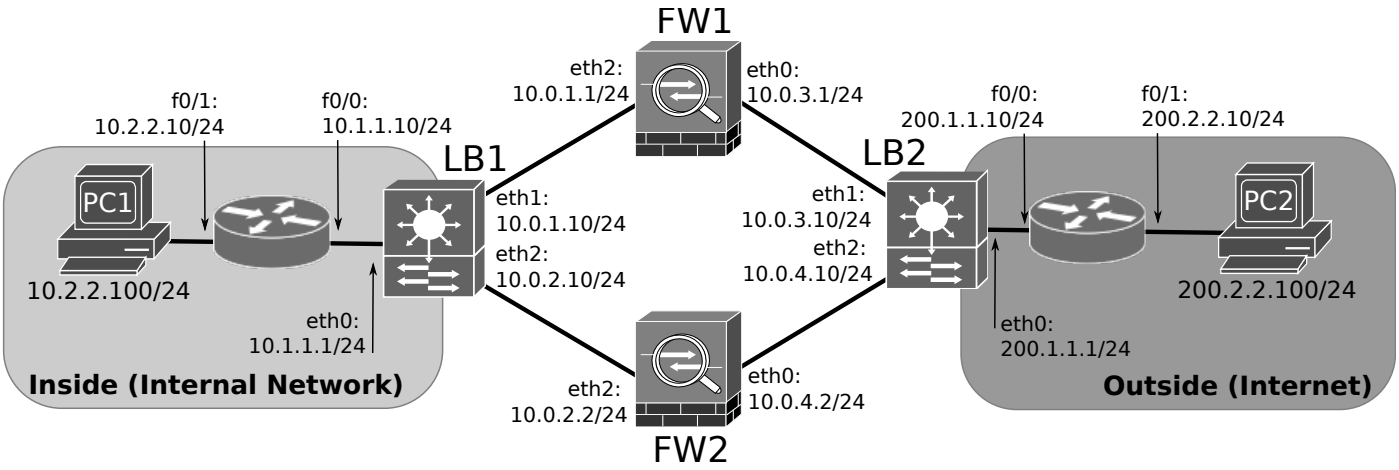
Assuming the existence of zones INSIDE and OUTSIDE and chains FROM-INSIDE-TO-OUTSIDEand FROM-OUTSIDE-TO-INSIDE:

```
...
# set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 action accept
# set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 protocol udp
# set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 destination port 5000-6000
...
# set firewall name FROM-OUTSIDE-TO-INSIDE rule 10 action accept
# set firewall name FROM-OUTSIDE-TO-INSIDE rule 10 state established enable
# set firewall name FROM-OUTSIDE-TO-INSIDE rule 10 state related enable
...
```

>> Test the connectivity between PC1 and PC2 using UDP pings: `PC1> ping 200.2.2.100 -P 17 -p 5001`

>> Verify the correct synchronization of the flow states between the firewalls.

Load-Balancing Scenario



6. Configure the network depicted in the previous figure using GNS3 with PC1 and PC2 as VPCS, and the the firewalls (FW1 and FW2) and Load-Balanceres (LB1 and LB2) as VyOS VM. Configure PCs addresses and gateways, Routers addresses and static routes, and Firewalls addresses, static routes and NAT/PAT mechanism. You must assume that the Internal Network as the 192.1.0.0/23 IPv4 public network.

Note that now the internal router default static route should be:

```
RouterInside(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1
```

Note: that the firewalls will not share state, therefore do not require a direct link between them.

7. Configure the Load-Balanceres names, addresses and static routes.

- For LB1:

```
# set system host-name LB1
# set interfaces ethernet eth0 address 10.1.1.1/24
# set interfaces ethernet eth1 address 10.0.1.10/24
# set interfaces ethernet eth2 address 10.0.2.10/24
# set protocols static route 10.2.2.0/24 next-hop 10.1.1.10
```

- For LB2:

```
# set system host-name LB2
# set interfaces ethernet eth0 address 200.1.1.1/24
# set interfaces ethernet eth1 address 10.0.3.10/24
# set interfaces ethernet eth2 address 10.0.4.10/24
# set protocols static route 200.2.2.0/24 next-hop 200.1.1.10
```

Note: missing routes (default and 192.1.0.0/23) will be implemented using the load balancing rules (see below).

>> Verify the configured addresses with: `$ show interfaces`

>> Verify the routing tables with: `$ show ip route`

8. Configure the Load-Balanceres load balancing service:

- For LB1:

```
# set load-balancing wan interface-health eth1 nexthop 10.0.1.1
# set load-balancing wan interface-health eth2 nexthop 10.0.2.2
# set load-balancing wan rule 1 inbound-interface eth0
# set load-balancing wan rule 1 interface eth1 weight 1
# set load-balancing wan rule 1 interface eth2 weight 1
# set load-balancing wan sticky-connections inbound
# set load-balancing wan disable-source-nat
```

- For LB2:

```
# set load-balancing wan interface-health eth1 nexthop 10.0.3.1
# set load-balancing wan interface-health eth2 nexthop 10.0.4.2
# set load-balancing wan rule 1 inbound-interface eth0
# set load-balancing wan rule 1 interface eth1 weight 1
# set load-balancing wan rule 1 interface eth2 weight 1
# set load-balancing wan sticky-connections inbound
# set load-balancing wan disable-source-nat
```

>> Test the connectivity between PC1 and PC2 using UDP pings for different UDP ports:

```
PC1> ping 200.2.2.100 -P 17 -p 5001
PC1> ping 200.2.2.100 -P 17 -p 5002
```

>> Start packet captures on the links that connect the firewalls and load balancers, and verify the correctness of the load balancing process between FW1 and FW2.

>> Inspect the status and active connections of the load balancing process:

```
$ show wan-load-balance status
$ show wan-load-balance connection
```

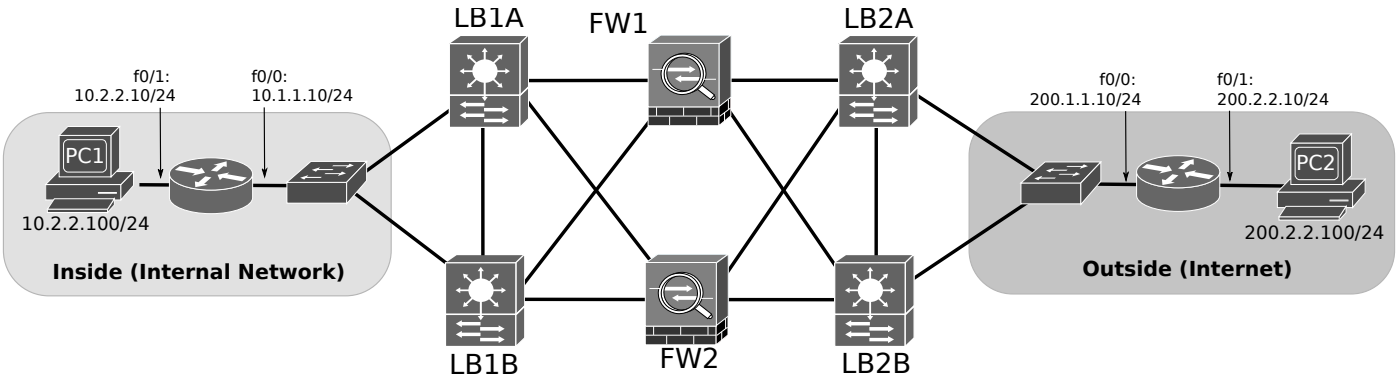
>> Using the following commands explain how VyOS implements the load balancing process using the iptables MANGLE table, and specific routing tables:

```
$ sudo iptables -vL -t mangle
$ ip rule show
$ ip route
$ ip route list table 201
$ ip route list table 202
```

Note: The VyOS load balancing process is a stateful process on a per-flow decision (command "set load-balancing wan sticky-connections inbound"). Upon the reception of a new inbound flow, a route is randomly chosen, and the choice is kept on the connection state table. This approach is not optimal for DDoS protection, ideally the decision should be based on a hash function (dependent of flow parameters - IP and/or ports) with no state kept in memory.

>> Propose new load-balancing algorithms.

Load-Balancing Scenario (with redundancy and state synchronization)



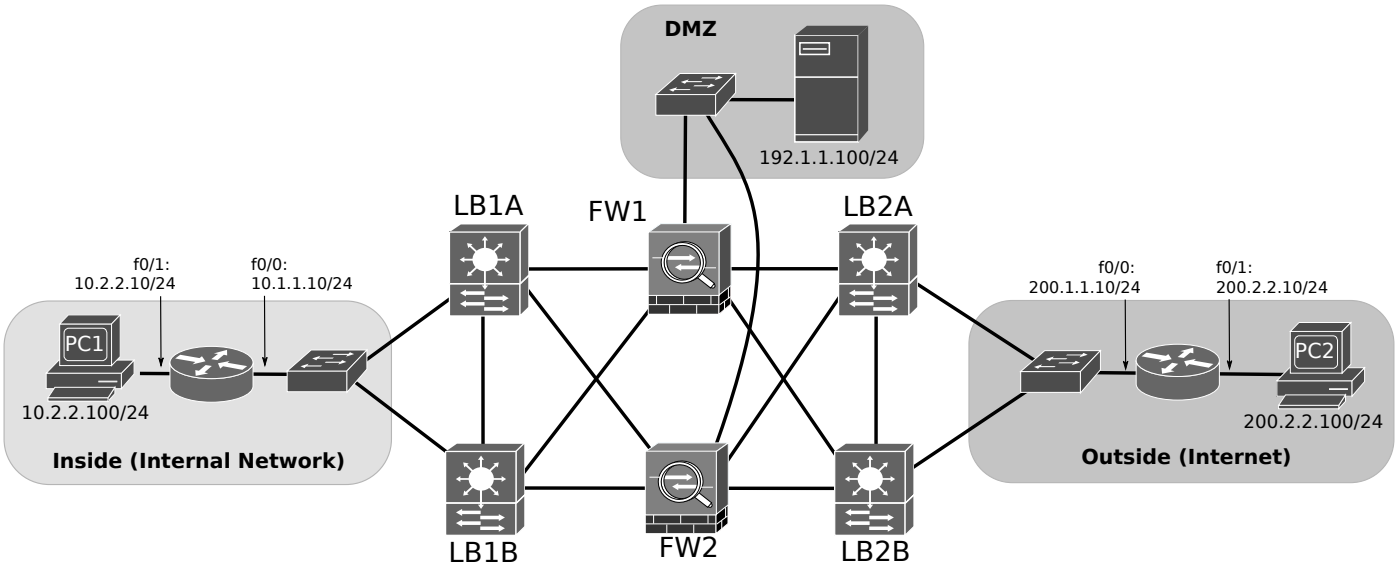
9. Configure the network depicted in the previous figure where the firewall load is distributed by redundant load-balancers (that share routing decisions). Activate conntrack-sync for the load-balancers.

>> Explain why the synchronization of the load-balancers allows the nonexistence of firewall synchronization.

>> Which load balancing algorithm may also allow the nonexistence of load-balancers synchronization?

>> Explain why device/connection states synchronization may be detrimental during a DDoS attack.

Policies Definition and Integrated Deployment



10. To the previous setup add a DMZ and deploy a server (may be simulated with a VPCS) that supports multiple services (e.g., HTTP, HTTPS, DNS, SSH, etc…).

>> Define a set of flow control policies that (i) incorporate the good practices of network defense, (ii) allow the internal users a limited access to internal/DMZ and external services, (iii) allow strict access to the public services available on the DMZ server(s), and (iv) block external users during a possible DDoS attack.

>> Deploy the appropriate firewall rules in the firewalls. You may assume that exists an external monitoring system that will identify the external IP address of the DDoS participants and dynamically provide an updated list.

>> (Extra) Create a script (running in a internal server directly connected to the firewalls) that automatically creates the blocking rules in the firewalls during a DDoS attack upon the identification of the attackers IP addresses. Suggestion: consider a remote connection via SSH to the firewalls.