HIGH-AVAILABILITY FIREWALLS SCENARIOS

University of Aveiro (*UA*)

Eduardo Lopes Fernandes,
João Afonso Pereira Ferreira

VERSION 1

Project 1

Security in Communications Networks

DETI

Mestrado de Engenharia de Computadores e Telemática (MECT)

Eduardo Lopes Fernandes,
João Afonso Pereira Ferreira

(102648) edu.fernandes@ua.pt,
(103037) ferreiraafonsojoao@ua.pt

07/04/2024

Índice

List of Figures

Chapter 1

Network Topology

For this project, it was developed the network depicted in the figure 1.1, where the firewall load is distributed by redundant load-balancers (that share routing decisions).

Overall, the project main goal aims to create a robust network architecture capable of effectively handling incoming traffic while ensuring high availability and security, despite component failures or malicious attacks.

This network is structured into three main zones: **INSIDE**, **OUTSIDE**, and **DMZ**. The inside zone is the internal network environment, containing a VPC (*10.2.2.100/24*), router, and switch1, which facilitates communication and data exchange within the organization s architecture. The OUTSIDE zone, Internet connectivity, hosts the *200.2.2.0/24* sub-network, featuring a VPCS to simulate an external device accessible from a company terminal.

The Network Address Translation (NAT) pool between FWs must be distinct because there is a small risk that both FWs will use the same IP port translation for different internal IPs, thus a division of pools was taken place (Fig. 1.2).

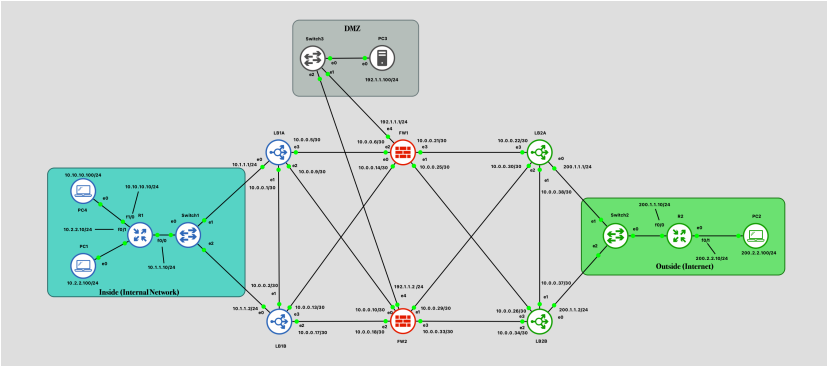

Figure 1.1: Network Topology

```
vyos@FW1:~$ show nat source translations
Pre-NAT              Post-NAT             Prot  Timeout
10.0.0.13            10.0.0.13            icmp  26
10.0.0.22            10.0.0.22            icmp  28
10.0.0.5             10.0.0.5             icmp  26
10.2.2.100           192.1.0.3            udp   26
```

```
vyos@FW2:~$ show nat source translations
Pre-NAT              Post-NAT             Prot  Timeout
10.0.0.34            10.0.0.34            icmp  26
10.0.0.17            10.0.0.17            icmp  25
10.0.0.30            10.0.0.30            icmp  25
10.0.0.9             10.0.0.9             icmp  27
```

Figure 1.2: NAT Translations

```
vyos@LB1A:~$ show conntrack table ipv4
TCP state codes: SS - SYN SENT, SR - SYN RECEIVED, ES - ESTABLISHED,
                 FW - FIN WAIT, CW - CLOSE WAIT, LA - LAST ACK,
                 TW - TIME WAIT, CL - CLOSE, LI - LISTEN

CONN ID    Source                Destination           Protocol        TIMEOUT
2031557721 10.0.0.2:37866        225.0.0.50:3780       udp [17]        29
3380751673 10.0.0.2              224.0.0.18            vrrp [112]      999
vyos@LB1A:~$ show wan-load-balance status
Chain WANLOADBALANCE_PRE (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ISP_eth2   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW statistic
mode random probability 0.50000000000
    0     0 ISP_eth3   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW
    0     0 CONNMARK   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           CONNMARK restore
```

```
vyos@LB1B:~$ show conntrack table ipv4
TCP state codes: SS - SYN SENT, SR - SYN RECEIVED, ES - ESTABLISHED,
                 FW - FIN WAIT, CW - CLOSE WAIT, LA - LAST ACK,
                 TW - TIME WAIT, CL - CLOSE, LI - LISTEN

CONN ID    Source                Destination           Protocol        TIMEOUT
3554481676 10.2.2.100:17622      200.2.2.100:5001      udp [17]        28
2535480990 10.0.0.1              224.0.0.18            vrrp [112]      451
vyos@LB1B:~$ show wan-load-balance status
Chain WANLOADBALANCE_PRE (1 references)
 pkts bytes target     prot opt in     out     source               destination
    3   252 ISP_eth2   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW statistic
mode random probability 0.50000000000
    2   168 ISP_eth3   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW
    0     0 CONNMARK   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           CONNMARK restore
```

```
vyos@LB2A:~$ show conntrack table ipv4
TCP state codes: SS - SYN SENT, SR - SYN RECEIVED, ES - ESTABLISHED,
                 FW - FIN WAIT, CW - CLOSE WAIT, LA - LAST ACK,
                 TW - TIME WAIT, CL - CLOSE, LI - LISTEN

CONN ID    Source                Destination           Protocol        TIMEOUT
1942392536 192.1.0.13:17622      200.2.2.100:5001      udp [17]        164
1884364731 192.1.0.3:17622       200.2.2.100:5001      udp [17]        166
2545441078 192.1.0.13:39255      200.2.2.100:5001      udp [17]        30
1486158769 10.0.0.37:56953       225.0.0.50:3780       udp [17]        29
vyos@LB2A:~$ show wan-load-balance status
Chain WANLOADBALANCE_PRE (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ISP_eth2   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW statistic
mode random probability 0.50000000000
    0     0 ISP_eth3   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW
   10   840 CONNMARK   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           CONNMARK restore
```

```
vyos@LB2B:~$ show conntrack table ipv4
TCP state codes: SS - SYN SENT, SR - SYN RECEIVED, ES - ESTABLISHED,
                 FW - FIN WAIT, CW - CLOSE WAIT, LA - LAST ACK,
                 TW - TIME WAIT, CL - CLOSE, LI - LISTEN

CONN ID    Source                Destination           Protocol        TIMEOUT
1738832688 10.0.0.38             224.0.0.18            vrrp [112]      599
1235779362 10.0.0.38:38612       225.0.0.50:3780       udp [17]        29
1557663133 192.1.0.3:17622       200.2.2.100:5001      udp [17]        161
3629818298 192.1.0.13:17622      200.2.2.100:5001      udp [17]        157
vyos@LB2B:~$ show wan-load-balance status
Chain WANLOADBALANCE_PRE (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ISP_eth2   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW statistic
mode random probability 0.50000000000
    0     0 ISP_eth3   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           state NEW
    0     0 CONNMARK   all  --  eth0   *       0.0.0.0/0            0.0.0.0/0           CONNMARK restore
```

Figure 1.3: Load Balancers Synchronization

1.1 Network Connectivity Testing and Analysis

1.1.1 Connectivity between PC1 and PC2

In order to test connectivity inside the network, the User Datagram Protocol (UDP) is used to enable communication between the INSIDE and OUTSIDE networks on ports 5000 to 6000 (Fig. 1.5). This service simulates situations in which UDP-based applications are needed for INSIDE devices to communicate with OUTSIDE devices. It s crucial to emphasise that communication cannot be initiated from the OUTSIDE network, as depicted in the figure 1.4. Despite the fact that it was possible to complete the successful ping between PC1 and PC2 (Fig. 1.5), the ping is not enough to debug and test its connectivity. Thus, with the help of *Wireshark* it was possible to confirm this connectivity testing on different links, with the filter "*udp*" (Fig. 1.6)

Routing decisions across redundant devices can be consistent when the load-balancers are synchronised. Efficient load balancing is ensured when *FW1* and *FW2* get traffic equally distributed via *LB1A* and *LB2A*. Both load-balancers are able to make informed routing decisions by staying synchronised and knowing the current state of the network. However, since the firewalls run separately, *firewall synchronisation is not required.* Firewalls do not need to exchange state information, since each manages its own traffic separately. The firewalls can function without synchronisation as long as the load-balancers distribute traffic equally (which is the case).

It was implemented a load balancing technique in our network settings, which

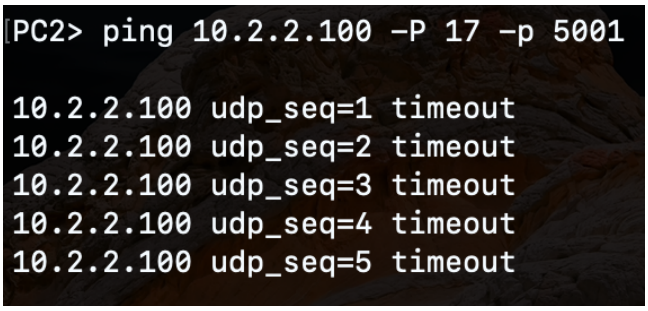

Figure 1.4: UDP Ping from PC2 to PC1

hashes the incoming packets through source IP address. By ensuring that packets with the same source IP address are consistently routed to the same firewall, this method efficiently distributes traffic throughout the network. Traffic is distributed equally as long as the load balancers share routing information and use the same hashing algorithm. (configured both as: *´set load-balancing wan rule 1 interface ethx weight 1´*)

Device or connection states synchronization may be detrimental during a Distributed Denial-of-Service (DDoS) attack as it can increase the attack s impact. A DDoS attack overloads the resources of the target system by flooding the network with excessive traffic. Devices must provide details about every connection they manage. This process of synchronisation can use more processing power and resources during a DDoS attack, making the attack more severe. It can also be simpler for attackers to take advantage of errors or flaws in the network infrastructure when connection statuses are synchronised. Given that every device is equipped with identical state information, meaning that if the attacker breaches one of those, the other that is synchronized will also be affected and breached.

Initially, it was a challenge to assure the synchronous between the load balancers. However, with the help of the teacher it was possible to verify the desired goal since the interfaces ethernet, that connect each firewall, configured previously must the same in each pair of balancers (E.x: LB1A, LB1B eth2 connects to firewall FW2) [Fig. 1.1].

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

84 bytes from 200.2.2.100 udp_seq=1 ttl=59 time=169.153 ms
84 bytes from 200.2.2.100 udp_seq=2 ttl=59 time=43.587 ms
84 bytes from 200.2.2.100 udp_seq=3 ttl=59 time=34.433 ms
84 bytes from 200.2.2.100 udp_seq=4 ttl=59 time=44.624 ms
84 bytes from 200.2.2.100 udp_seq=5 ttl=59 time=42.678 ms

PC1> ping 200.2.2.100 -P 17 -p 5999

84 bytes from 200.2.2.100 udp_seq=1 ttl=59 time=40.227 ms
84 bytes from 200.2.2.100 udp_seq=2 ttl=59 time=39.835 ms
84 bytes from 200.2.2.100 udp_seq=3 ttl=59 time=37.563 ms
84 bytes from 200.2.2.100 udp_seq=4 ttl=59 time=39.151 ms
84 bytes from 200.2.2.100 udp_seq=5 ttl=59 time=39.010 ms

PC1> ping 200.2.2.100 -P 17 -p 6001

200.2.2.100 udp_seq=1 timeout
200.2.2.100 udp_seq=2 timeout
200.2.2.100 udp_seq=3 timeout
200.2.2.100 udp_seq=4 timeout
200.2.2.100 udp_seq=5 timeout

PC1> ping 200.2.2.100 -P 17 -p 4999

200.2.2.100 udp_seq=1 timeout
200.2.2.100 udp_seq=2 timeout
200.2.2.100 udp_seq=3 timeout
200.2.2.100 udp_seq=4 timeout
200.2.2.100 udp_seq=5 timeout
```

Figure 1.5: UDP Ping from PC1 to PC2 utilizing a range of ports between 5000 and 6000

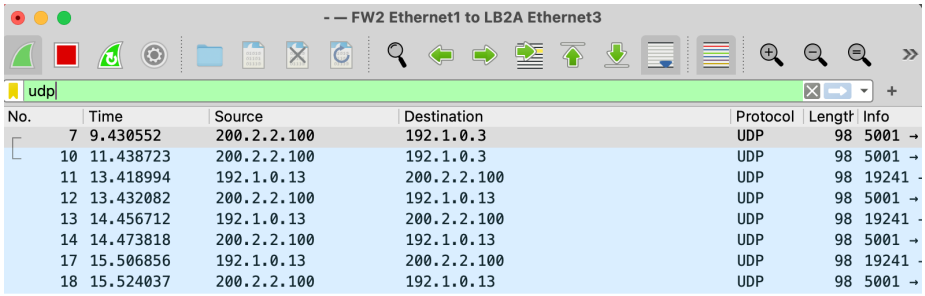

Figure 1.6: UDP Ping from PC1 to PC2 - Wireshark - Link (FW2 – LB2A), 2 way captured packets

Chapter 2

Zones Definition, Inter-Zone Rules and Policies

2.1 Zone INSIDE

In this zone, two main policies were implemented, namely **FROM-DMZ-TO-INSIDE** and **FROM-INSIDE-TO-OUTSIDE**. These policies are intended to manage and secure traffic entering and exiting the INSIDE zone, as well as into the DMZ.

2.1.1 FROM-INSIDE-TO-OUTSIDE

The **FROM-INSIDE-TO-OUTSIDE** Policy permits certain types of traffic from the INSIDE zone to access the OUTSIDE zone.

- **Rule** 10 allows UDP communication on ports 5000-6000.
- **Rule** 20 allows ICMP Echo Requests (type 8) to be used for connectivity testing and monitoring.

2.1.2 FROM-INSIDE-TO-DMZ

The **FROM-INSIDE-TO-DMZ** Policy regulates traffic from the INSIDE zone to the DMZ, providing regulated access to DMZ services and systems.

- **Rule** 10 enables ICMP Echo Requests (type 8) to addresses inside the 192.1.1.0/24 subnet. This is critical for network diagnostics and administration across these two zones.

2.2 Zone OUTSIDE

For this zone, two main policies were implemented, namely **FROM-OUTSIDE-TO-INSIDE** and **FROM-OUTSIDE-TO-DMZ**. These policies are designed to control and secure traffic moving from external sources into the INSIDE zone and the DMZ, respectively.

2.2.1 FROM-OUTSIDE-TO-INSIDE

The FROM-OUTSIDE-TO-INSIDE Policy manages incoming traffic from the OUTSIDE zone to the INSIDE zone.

- **Rule** 10 restricts access to the INSIDE zone to existing or connected connections from external sources.

2.2.2 FROM-OUTSIDE-TO-DMZ

The FROM-OUTSIDE-TO-DMZ Policy regulates traffic from the OUTSIDE zone to the DMZ.

- **Rule** 12 allows UDP communication destined for port 8080 to a specified address within the DMZ (192.1.1.140). This rule makes it easier for external users to access a specific DMZ-hosted service.

2.3 Zone DMZ

On DMZ zone, it was deployed a server (simulated with a Virtual PC Simulator (VPC)) that supports the following services HTTP, HTTPS, DNS, SSH, as it is possible to verify on the next configuration of the two firewalls. For *DMZ Zone* it was added a VPC (PC3), where strictly the two firewalls (FW1 and FW2) can send traffic to the DMZ.

```
name FROM-INSIDE-TO-DMZ name FROM-INSIDE-TO-DMZ {
    default-action drop
    ... (ICMP)
    rule 20 {
        action accept
        description "Accept HTTP, HTTPS and SSH from Admin to DMZ"
        destination {
            address 192.1.1.0/24
            port 22,80,443
        }
        protocol tcp
        source {
            address 10.10.10.0/24
        }
    }

```

```
    rule 30 {
        action accept
        description "Allow DNS access to DMZ"
        destination {
            address 192.1.1.0/24
            port 53
        }
        protocol udp
    }
    rule 40 {
        action accept
        description "Accept HTTP and HTTPS from INSIDE to DMZ"
        destination {
            address 192.1.1.0/24
            port 80,443
        }
        protocol tcp
    }
}
```

2.3.1 DNS Server

The firewall rule described within the previous configuration allows UDP traffic destined for *port 53* to enter the DMZ zone. The DMZ zone has the IP address range 192.1.1.0/24, which includes the assigned **Domain Name System (DNS) server** at 192.1.1.100 (a VPC in this case). Thus, as specified in the configuration, the firewall should allow clear passage of DNS packets to the DNS server within the DMZ (Fig. 2.1).

If DNS traffic fails to traverse the network as expected, it is recommended to examine the routing and network parameters [1]. The purpose of this investigation is to ensure that network traffic is accurately directed to the DMZ zone, as well as to identify and resolve any potential barriers to DNS packet transport.

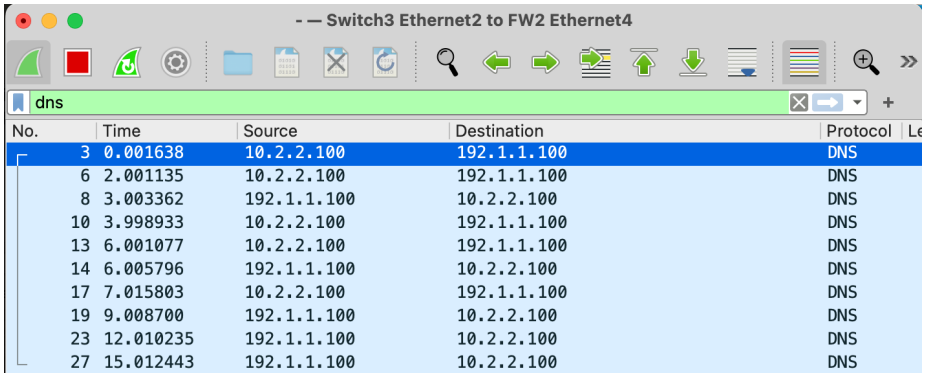

Figure 2.1: UDP Ping from PC1 to PC3 (DNS Server) using port 53

2.3.2 SSH

The firewall rule described within the previous configuration allows UDP traffic destined for *port 22* to enter the DMZ zone. In the standard configuration, **Secure Shell (SSH)** communicates exclusively over Transmission Control Protocol (TCP) across an IP network (using port 22). However, for the purposes of, somehow, not being able to successfully ping through protocol tcp (firewalls are blocking the traffic on the response, as it is possible to observe on the fig. 2.3, the packet exchange is happening between switch3 and PC4) its fundamentals, UDP was selected to investigate the consequences and potential behaviours of running session-oriented protocols over an unreliable communication layer like UDP .

The main goal of using *SSH* to control traffic from the Inside Zone to the DMZ is to create a secure and encrypted channel for remote administration and management of devices in the DMZ. By allowing SSH communication from the Inside zone to the DMZ, administrators can securely access and administer servers, applications, and other resources housed in the DMZ from within the internal network.

Additionally, conducting a ping test from PC1 (10.2.2.100), located in the *Inside zone*, to PC4 (192.1.1.100) in the *DMZ zone* as well as observing the UDP packets on wireshark, served as a method to validate the effectiveness and exclusivity of the SSH connection. By executing the command ping 192.1.1.100 -P 17 -p 22 -w 2000 , which sends requests to the SSH port (port 22) of PC4 using UDP , it was possible to verify that only authorized SSH traffic was permitted from the Inside zone to the DMZ, assuring exclusivity (Fig. 2.2). This verification ensures that the SSH connection is properly secured, maintaining the most valuable keys that are integrity and confidentiality between the two zones.

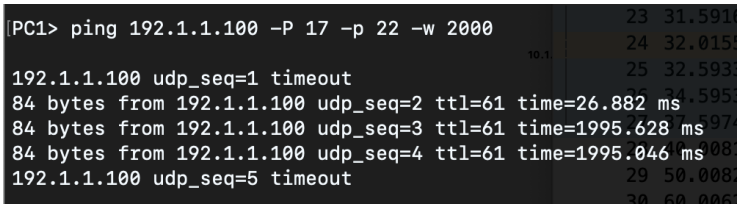

Figure 2.2: UDP Ping from PC1 to PC3 (SSH) using port 22

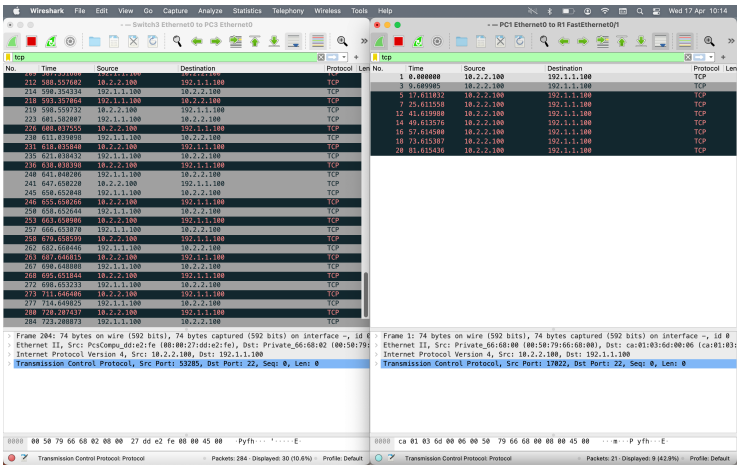

Figure 2.3: TCP Ping from PC1 to PC3 (SSH) using port 22

2.3.3 DMZ IP Blocking

This subsection addresses the implementation and testing of **IP blocking** within the DMZ of the specified network infrastructure. The DMZ acts as a key barrier between the internal network and the untrustworthy external connections. As part of the chosen security procedures, it was added a new VPC to the inner zone, which allows to examine and restrict access to the DMZ from certain sources. Specifically, it was used IP blocking rules to limit traffic from specific IP addresses(as a case of study, in this case, from the ip 10.10.10.10/24). Thus, by using this method, it is possible to improve network security and well prepared against possible threats.

```
name FROM-VPC-TO-DMZ {
    default-action accept
    rule 5 {
    action drop
    destination {
        address 192.1.1.0/24
    }
    icmp {
        type 8
    }
    protocol icmp
    source {
        address 10.10.10.10/24
    }
    }
}
```

In order to conduct a test of the intended blocking and its correct configuration, two pings were performed, one from PC1 (10.2.2.100/24) and from the new PC4 (10.10.10.10/24). As it is possible to observe in the Fig. 2.4, it is possible to ping from PC1 to the DMZ PC3, however, from PC4 it is not, since the firewalls have a rule that is blocking ICMP traffic from this pc, which is what it was pretended.

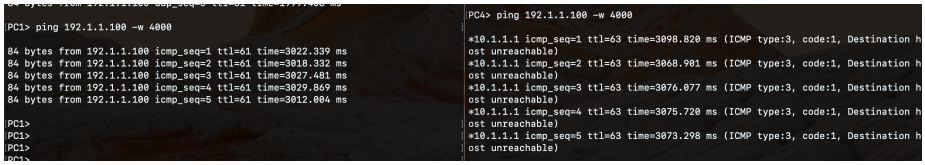

Figure 2.4: Blocking traffic from PC4 to DMZ

Chapter 3

Conclusion

Concluding, it was developed a robust network architecture capable of effectively handling incoming traffic while ensuring high availability and security, according to the information and material provided by the teachers of "'Security in Communications Networks" [2].

Acronyms

MECT Mestrado de Engenharia de Computadores e Telemática

NAT Network Address Translation

UDP User Datagram Protocol

TCP Transmission Control Protocol

VPC Virtual PC Simulator

DDoS Distributed Denial-of-Service

DNS Domain Name System

SSH Secure Shell

Bibliography

[1] DigiCert, *What is a global dns network?* https://dnsmadeeasy.com/post/global-dns-network-benefits, Accessed: April 2024, 2023.

[2] SRC - UA, *Consulted pdfs provided by the teacher*, https://elearning.ua.pt/course/view.php?id=16070, Accessed: 2nd semester 2024, 2024.

Chapter 4

Appendix

4.1 Load Balancer Configuration

4.1.1 LB1A Configuration

```
set system host-name LB1A
set high-availability vrrp group LB1Cluster vrid 1
set high-availability vrrp group LB1Cluster interface eth1
set high-availability vrrp group LB1Cluster virtual-address 10.0.0.1/30
set high-availability vrrp sync-group LB1Cluster member LB1Cluster
set high-availability vrrp group LB1Cluster rfc3768-compatibility

set interfaces ethernet eth0 address 10.1.1.1/24
set interfaces ethernet eth1 address 10.0.0.1/30
set interfaces ethernet eth2 address 10.0.0.9/30
set interfaces ethernet eth3 address 10.0.0.5/30


set load-balancing wan interface-health eth2 nexthop 10.0.0.6
set load-balancing wan interface-health eth3 nexthop 10.0.0.10
set load-balancing wan rule 1 inbound-interface eth0
set load-balancing wan rule 1 interface eth2 weight 1
set load-balancing wan rule 1 interface eth3 weight 1
set load-balancing wan rule 1 protocol all
set load-balancing wan disable-source-nat

set protocols static route 10.2.2.0/24 next-hop 10.1.1.10

set service conntrack-sync accept-protocol tcp,udp,icmp
set service conntrack-sync failover-mechanism vrrp sync-group LB1Cluster
set service conntrack-sync interface eth1
set service conntrack-sync mcast-group 225.0.0.50
set service conntrack-sync disable-external-cache
```

4.1.2 LB1B Configuration

```
set system host-name LB1B
set high-availability vrrp group LB1Cluster vrid 1
set high-availability vrrp group LB1Cluster interface eth1
set high-availability vrrp group LB1Cluster virtual-address 10.0.0.1/30
set high-availability vrrp sync-group LB1Cluster member LB1Cluster
set high-availability vrrp group LB1Cluster rfc3768-compatibility

set interfaces ethernet eth0 address 10.1.1.2/24
set interfaces ethernet eth1 address 10.0.0.2/30
set interfaces ethernet eth2 address 10.0.0.17/30
set interfaces ethernet eth3 address 10.0.0.13/30

set load-balancing wan interface-health eth2 nexthop 10.0.0.18
set load-balancing wan interface-health eth3 nexthop 10.0.0.14
set load-balancing wan rule 1 inbound-interface eth0
set load-balancing wan rule 1 interface eth2 weight 1
set load-balancing wan rule 1 interface eth3 weight 1
set load-balancing wan rule 1 protocol all
set load-balancing wan disable-source-nat

set protocols static route 0.0.0.0/0 next-hop 10.0.0.14
set protocols static route 0.0.0.0/0 next-hop 10.0.0.18
set protocols static route 10.0.0.0/30 next-hop 10.0.0.1
set protocols static route 10.0.0.12/30 next-hop 10.0.0.14
set protocols static route 10.0.0.16/30 next-hop 10.0.0.18
set protocols static route 10.2.2.0/24 next-hop 10.1.1.10

set service conntrack-sync accept-protocol tcp,udp,icmp
set service conntrack-sync failover-mechanism vrrp sync-group LB1Cluster
set service conntrack-sync interface eth1
set service conntrack-sync mcast-group 225.0.0.50
set service conntrack-sync disable-external-cache
```

4.1.3 LB2A Configuration

```
set system host-name LB2A
set high-availability vrrp group LB2Cluster vrid 2
set high-availability vrrp group LB2Cluster interface eth1
set high-availability vrrp group LB2Cluster virtual-address
    192.168.100.1/24
set high-availability vrrp sync-group LB2Cluster member LB2Cluster
set high-availability vrrp group LB2Cluster rfc3768-compatibility

set interfaces ethernet eth0 address 200.1.1.1/24
set interfaces ethernet eth1 address 10.0.0.38/30
set interfaces ethernet eth2 address 10.0.0.30/30
```

```
set interfaces ethernet eth3 address 10.0.0.22/30

set load-balancing wan interface-health eth2 nexthop 10.0.0.29
set load-balancing wan interface-health eth3 nexthop 10.0.0.21
set load-balancing wan rule 1 inbound-interface eth0
set load-balancing wan rule 1 interface eth2 weight 1
set load-balancing wan rule 1 interface eth3 weight 1
set load-balancing wan rule 1 protocol all
set load-balancing wan sticky-connections inbound
set load-balancing wan disable-source-nat

set protocols static route 200.2.2.0/24 next-hop 200.1.1.10

set service conntrack-sync accept-protocol tcp,udp,icmp
set service conntrack-sync failover-mechanism vrrp sync-group LB2Cluster
set service conntrack-sync interface eth1
set service conntrack-sync mcast-group 225.0.0.50
set service conntrack-sync disable-external-cache
```

4.1.4 LB2B Configuration

```
set system host-name LB2B
set high-availability vrrp group LB2Cluster vrid 2
set high-availability vrrp group LB2Cluster interface eth1
set high-availability vrrp group LB2Cluster virtual-address
    192.168.100.1/24
set high-availability vrrp sync-group LB2Cluster member LB2Cluster
set high-availability vrrp group LB2Cluster rfc3768-compatibility

set interfaces ethernet eth0 address 200.1.1.2/24
set interfaces ethernet eth1 address 10.0.0.37/30
set interfaces ethernet eth2 address 10.0.0.34/30
set interfaces ethernet eth3 address 10.0.0.26/30

set load-balancing wan interface-health eth2 nexthop 10.0.0.33
set load-balancing wan interface-health eth3 nexthop 10.0.0.25
set load-balancing wan rule 1 inbound-interface eth0
set load-balancing wan rule 1 interface eth2 weight 1
set load-balancing wan rule 1 interface eth3 weight 1
set load-balancing wan rule 1 protocol all
set load-balancing wan disable-source-nat

set protocols static route 200.2.2.0/24 next-hop 200.1.1.10
set service conntrack-sync accept-protocol tcp,udp,icmp
set service conntrack-sync interface eth1
set service conntrack-sync failover-mechanism vrrp sync-group LB2Cluster
set service conntrack-sync mcast-group 225.0.0.50
set service conntrack-sync disable-external-cache
```

4.2 Firewall Configuration

4.2.1 FW1 Configuration

```
set system host-name FW1
# FROM-INSIDE-TO-DMZ policy
set firewall name FROM-INSIDE-TO-DMZ default-action drop
set firewall name FROM-INSIDE-TO-DMZ rule 10 action accept
set firewall name FROM-INSIDE-TO-DMZ rule 10 description "Accept ICMP
    Echo Request"
set firewall name FROM-INSIDE-TO-DMZ rule 10 destination address
    192.1.1.0/24
set firewall name FROM-INSIDE-TO-DMZ rule 10 icmp type 8
set firewall name FROM-INSIDE-TO-DMZ rule 10 protocol icmp
set firewall name FROM-INSIDE-TO-DMZ rule 20 action accept
set firewall name FROM-INSIDE-TO-DMZ rule 20 description "Accept SSH (
    TCP) from Admin to DMZ"
set firewall name FROM-INSIDE-TO-DMZ rule 20 destination address
    192.1.1.0/24
set firewall name FROM-INSIDE-TO-DMZ rule 20 destination port 22
set firewall name FROM-INSIDE-TO-DMZ rule 20 protocol tcp
set firewall name FROM-INSIDE-TO-DMZ rule 30 action accept
set firewall name FROM-INSIDE-TO-DMZ rule 30 description "Allow DNS
    access to DMZ"
set firewall name FROM-INSIDE-TO-DMZ rule 30 destination address
    192.1.1.0/24
set firewall name FROM-INSIDE-TO-DMZ rule 30 destination port 53
set firewall name FROM-INSIDE-TO-DMZ rule 30 protocol udp

# FROM-INSIDE-TO-OUTSIDE policy
set firewall name FROM-INSIDE-TO-OUTSIDE default-action drop
set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 action accept
set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 destination port
    5000-6000
set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 protocol udp
set firewall name FROM-INSIDE-TO-OUTSIDE rule 20 action accept
set firewall name FROM-INSIDE-TO-OUTSIDE rule 20 description "Accept
    ICMP Echo Request"
set firewall name FROM-INSIDE-TO-OUTSIDE rule 20 icmp type 8
set firewall name FROM-INSIDE-TO-OUTSIDE rule 20 protocol icmp

# FROM-OUTSIDE-TO-DMZ policy
set firewall name FROM-OUTSIDE-TO-DMZ default-action drop
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 action accept
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 description "Allow ICMP
    Echo Request to DMZ"
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 destination address
    192.1.1.0/24
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 icmp type 8
```

```
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 protocol icmp
set firewall name FROM-OUTSIDE-TO-DMZ rule 11 action accept
set firewall name FROM-OUTSIDE-TO-DMZ rule 11 description "Allow DNS
    access to DMZ"
set firewall name FROM-OUTSIDE-TO-DMZ rule 11 destination address
    192.1.1.0/24
set firewall name FROM-OUTSIDE-TO-DMZ rule 11 destination port 53
set firewall name FROM-OUTSIDE-TO-DMZ rule 11 protocol udp

# FROM-OUTSIDE-TO-INSIDE policy
set firewall name FROM-OUTSIDE-TO-INSIDE default-action drop
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

# FROM-VPC-TO-DMZ policy
set firewall name FROM-VPC-TO-DMZ default-action drop
set firewall name FROM-VPC-TO-DMZ rule 5 action drop
set firewall name FROM-VPC-TO-DMZ rule 5 description "Block UDP traffic
    from VPC to DMZ"
set firewall name FROM-VPC-TO-DMZ rule 5 destination address 192.1.1.100
set firewall name FROM-VPC-TO-DMZ rule 5 destination port 53
set firewall name FROM-VPC-TO-DMZ rule 5 protocol udp
set firewall name FROM-VPC-TO-DMZ rule 5 source address 10.10.10.100

# TO-INSIDE policy
set firewall name TO-INSIDE default-action drop
set firewall name TO-INSIDE rule 20 action accept
set firewall name TO-INSIDE rule 20 description "Accept Established-
    Related Connections"
set firewall name TO-INSIDE rule 20 state established enable
set firewall name TO-INSIDE rule 20 state related enable

# Configure interface addresses
set interfaces ethernet eth0 address 10.0.0.14/30
set interfaces ethernet eth1 address 10.0.0.25/30
set interfaces ethernet eth2 address 10.0.0.6/30
set interfaces ethernet eth3 address 10.0.0.21/30
set interfaces ethernet eth4 address 192.1.1.1/24

# Configure NAT rules
set nat source rule 10 outbound-interface eth3
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
set nat source rule 20 outbound-interface eth1
set nat source rule 20 source address 10.0.0.0/8
set nat source rule 20 translation address 192.1.0.1-192.1.0.10

# Configure static routes
```

```
set protocols static route 0.0.0.0/0 next-hop 10.0.0.22
set protocols static route 0.0.0.0/0 next-hop 10.0.0.26
set protocols static route 10.0.0.0/12 next-hop 10.0.0.5
set protocols static route 10.0.0.0/12 next-hop 10.0.0.13
set protocols static route 10.0.0.0/24 next-hop 10.0.0.5
set protocols static route 10.0.0.0/24 next-hop 10.0.0.13

# Configure zone policies
set zone-policy zone DMZ default-action drop
set zone-policy zone DMZ description "DMZ (Server Farm)"
set zone-policy zone DMZ from INSIDE firewall name FROM-INSIDE-TO-DMZ
set zone-policy zone DMZ from OUTSIDE firewall name FROM-OUTSIDE-TO-DMZ
set zone-policy zone DMZ interface eth4

set zone-policy zone INSIDE default-action drop
set zone-policy zone INSIDE description "Inside (Internal Network)"
set zone-policy zone INSIDE from DMZ firewall name TO-INSIDE
set zone-policy zone INSIDE from OUTSIDE firewall name TO-INSIDE
set zone-policy zone INSIDE interface eth0
set zone-policy zone INSIDE interface eth2

set zone-policy zone OUTSIDE default-action drop
set zone-policy zone OUTSIDE description "Outside (Internet)"
set zone-policy zone OUTSIDE from INSIDE firewall name FROM-INSIDE-TO-
    OUTSIDE
set zone-policy zone OUTSIDE interface eth1
set zone-policy zone OUTSIDE interface eth3
```

4.2.2 FW2 Configuration

```
set system host-name FW2

# Inter-zone configuration pretty much the same as well as firewall
# policies (changing the needed interfaces)
set interfaces ethernet eth0 address 10.0.0.10/30
set interfaces ethernet eth1 address 10.0.0.29/30
set interfaces ethernet eth2 address 10.0.0.18/30
set interfaces ethernet eth3 address 10.0.0.33/30
set interfaces loopback lo

set nat source rule 10 outbound-interface eth3
set nat source rule 10 source address 10.0.0.0/8
set nat source rule 10 translation address 192.1.0.11-192.1.0.20

set nat source rule 20 outbound-interface eth1
set nat source rule 20 source address 10.0.0.0/8
set nat source rule 20 translation address 192.1.0.11-192.1.0.20

```

```
set protocols static route 0.0.0.0/0 next-hop 10.0.0.30
set protocols static route 0.0.0.0/0 next-hop 10.0.0.34
set protocols static route 10.0.0.0/12 next-hop 10.0.0.9
set protocols static route 10.0.0.0/12 next-hop 10.0.0.17
set protocols static route 10.0.0.0/24 next-hop 10.0.0.9
set protocols static route 10.0.0.0/24 next-hop 10.0.0.17
```