

A imagem acima descreve a topologia desenvolvida, assim como os endereços IP utilizados para o funcionamento total da rede. Verifica-se a criação de zonas, de maneira a criar limites lógicos entre diferentes partes de uma rede e com o intuito de aplicar políticas específicas a cada zona para controle de tráfego. Foram definidas três zonas: Inside, Outside e DMZ, mais à frente, explicaremos cada uma.

Load Balancing e Sincronização

É o processo de distribuir o trabalho a realizar por vários recursos que possam computar este trabalho. Assim sendo, o mesmo se aplica às firewalls, onde o tráfego é distribuído por várias firewalls, de maneira a garantir que este é processado eficientemente. Firewalls com load balancing possuem várias vantagens, sendo a performance melhorada uma consequência da distribuição de trabalho, evitando também que alguma firewall se torne um bottleneck. Com o load balancing, obtém-se também uma disponibilidade maior, sendo que, caso uma firewall falhe, o load balance consegue redirecionar o tráfego para uma firewall de backup, de maneira a garantir que o tráfego continua a ser processado. A redundância obtida e a segurança conseguida devido à aplicação de políticas diferentes a segmentos de redes diferentes são o foco da nossa análise.

Na nossa rede, as firewalls são stateful, o que significa que aplicam regras ao fluxo geral de comunicações e não só a pacotes. Analisando a maneira como o load balancing foi implementado na nossa rede, percebemos que os load balancers escolhem para que interface mandam um pacote com um dado destino e a partir daí todos os pacotes com a mesma origem e destino vão ser reencaminhados por essa interface. Desta maneira não é necessário haver sincronização entre firewalls, uma vez que sempre que uma fonte envia algo para um certo destino, vai passar sempre na mesma firewall, devido ao load balancing implementado. A configuração no nosso caso de load balancing para dois interfaces:

```
set load-balancing wan interface-health eth1 nexthop 10.1.7.1
set load-balancing wan interface-health eth2 nexthop 10.1.9.2
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
```

Ainda nesta parte, consideramos importante referir que existem algoritmos de load balancing que permitem determinar que um server/firewall receberá um certo request. Exemplo disso é o IP Hash, em que o IP de origem e de destino dos pacotes é usado para determinar o alvo do pedido, através de um output de Hash Functions . Este algoritmo não necessita de sincronização de estados quer a nível de firewall ou de load balancing, sendo que cada load balancer

consegue calcular independentemente qual deve ser o alvo de cada pedido independentemente. Por outro lado, algoritmos como o Round-Robin ou Least Connections necessitam que os load balancers mantenham o estado partilhado, o que pode levar a problemas de sincronização.

CONN ID	Source	Destination	Protocol	TIMEOUT
2068215489	10.2.2.100:59667	192.1.1.50:22	tcp [6] SS	93
4109790480	10.2.2.100:11152	200.2.2.100:59552	tcp [6] SS	63
2072895974	10.0.10.2:57997	225.0.0.50:3780	udp [17]	29

Foi usado conntrack para a sincronização dos nossos load-balancers, estando acima descrita a tabela do LB1A a representada. Nas primeiras duas linhas podemos observar uma tentativa de comunicação da parte da rede Inside e na terceira linha a rede para comunicação e sincronização entre os load-balancers.

Como na nossa topologia, os load balancers são stateful, uma vez que guardam o estado de escolha de rotas e partilham e sincronizam esse estado com outros load-balancers, e o algoritmo de load balancing usado (round-robin), achamos importante falar de cenários em que manter esta sincronização pode ser detrimental, como por exemplo, durante um ataque DDOS. Neste caso, a sincronização torna-se um problema devido ao consumo de recursos, uma vez que sincronizar estados de dispositivos e de conexões em redes com vários nós torna-se algo que consome muitos muita memória. Isto pode ser aproveitado num ataque DDOS, uma vez que esta carga no sistema pode criar um bottleneck e reduzir a performance do sistema.

Por outro lado, se o mecanismo de sincronização não tiver capacidade para lidar com grandes quantidades de tráfego, este pode tornar-se um alvo para os atacantes amplificarem o ataque. Da mesma maneira, se ocorre um erro de sincronização devido a tráfego excessivo, o sistema pode tornar-se mais vulnerável ao ataque, se não conseguir recuperar rapidamente o suficiente para continuar a funcionar.

Zonas, Políticas e aplicação de regras nas Firewalls

```
set zone-policy zone INSIDE description "Inside (Internal Network)"
set zone-policy zone INSIDE interface eth1
set zone-policy zone INSIDE interface eth2
set zone-policy zone OUTSIDE description "Outside (Internet)"
set zone-policy zone OUTSIDE interface eth3
set zone-policy zone OUTSIDE interface eth4
set zone-policy zone DMZ description "DMZ"
set zone-policy zone DMZ interface eth5
```

A imagem acima representa os comandos usados para definir as zonas, como mostrado na topologia previamente referenciada. A zona INSIDE representa a rede privada, a zona OUTSIDE representa os dispositivos externos (que se encontram na Internet. Vemos também a definição de uma zona DMZ, área partilhada por ambas as redes anteriores, mas que providencia serviços exclusivos à rede interna. Aplicaremos regras de controlo de fluxo sobre estas mesmas zonas de forma a:

- A zona INSIDE usufrui dos serviços de FTP, SSH e SMTP da DMZ. Pode comunicar por HTTP, HTTPS ou Streaming (portos com 64936 até 64939) com a internet geral
- A zona OUTSIDE usufrui também de FTP e SMTPS (porta diferente de SMTP), da DMZ, mas também consegue aceder ao serviço de DNS externo. Esta zona não inicia conexões com a rede privada, mas pode fornecer respostas
- A zona da DMZ não inicia conexões, apenas fornece respostas aos pedidos que recebe. Esta zona tem um endereço reservado às suas funcionalidades para a rede privada (192.1.1.50) e outro para as funcionalidades disponíveis a todos (192.1.1.100)

Por questões práticas, para verificação do correto funcionamento do fluxo de informação, foram usados pings TCP e UDP em portas específicas entre os VPCS localizados em cada zona. Foi também considerado o endereço 192.142.44.18 como malicioso, sendo este bloqueado de qualquer comunicação, simulando a prevenção de ataques DDOS. Para prevenir spoofing à rede interna, foi bloqueada a entrada na mesma de qualquer pacote com o endereço igual ao da rede interna. É também de considerar o uso de NAT para tradução de endereços privados na pool de endereços públicos

192.1.0.1-192.1.0.20. Segue no **Anexo I** as configurações relevantes nas firewalls.

ANEXO I

```
set zone-policy zone INSIDE description "Inside (Internal Network)"
```

```
set zone-policy zone INSIDE interface eth1
```

```
set zone-policy zone INSIDE interface eth2
```

```
set zone-policy zone OUTSIDE description "Outside (Internet)"
```

```
set zone-policy zone OUTSIDE interface eth3
```

```
set zone-policy zone OUTSIDE interface eth4
```

```
set zone-policy zone DMZ description "DMZ"
```

```
set zone-policy zone DMZ interface eth5
```

```
commit
```

```
set firewall name FROM-INSIDE-TO-OUTSIDE rule 10 description "inside network access for streaming ports through UDP"
```

```
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 64936-64939
```

```
set firewall name FROM-INSIDE-TO-OUTSIDE rule 11 description "inside network access for HTTP and HTTPS through TCP"
```

```
set firewall name FROM-INSIDE-TO-OUTSIDE rule 11 action accept
```

```
set firewall name FROM-INSIDE-TO-OUTSIDE rule 11 protocol tcp
```

```
set firewall name FROM-INSIDE-TO-OUTSIDE rule 11 destination port 80,443
```

```
set firewall name FROM-OUTSIDE-TO-INSIDE rule 3 action drop
```

```
set firewall name FROM-OUTSIDE-TO-INSIDE rule 3 protocol all
```

```
set firewall name FROM-OUTSIDE-TO-INSIDE rule 3 source address 10.2.2.0/24
set firewall name FROM-OUTSIDE-TO-INSIDE rule 4 description "block this blacklisted attacker"
set firewall name FROM-OUTSIDE-TO-INSIDE rule 4 action drop
set firewall name FROM-OUTSIDE-TO-INSIDE rule 4 protocol all
set firewall name FROM-OUTSIDE-TO-INSIDE rule 4 source address 195.142.44.18
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
set firewall name FROM-OUTSIDE-TO-INSIDE rule 10 destination address 192.1.0.1-192.1.0.20
commit
```

```
set firewall name FROM-INSIDE-TO-DMZ rule 10 description "inside network access to SSH"
set firewall name FROM-INSIDE-TO-DMZ rule 10 action accept
set firewall name FROM-INSIDE-TO-DMZ rule 10 protocol tcp
set firewall name FROM-INSIDE-TO-DMZ rule 10 destination address 192.1.1.50
set firewall name FROM-INSIDE-TO-DMZ rule 10 destination port 22
set firewall name FROM-INSIDE-TO-DMZ rule 12 description "inside network access to ftp and smtp"
set firewall name FROM-INSIDE-TO-DMZ rule 12 action accept
set firewall name FROM-INSIDE-TO-DMZ rule 12 protocol tcp
set firewall name FROM-INSIDE-TO-DMZ rule 12 destination address 192.1.1.100
set firewall name FROM-INSIDE-TO-DMZ rule 12 destination port 20,25
```

```
set firewall name FROM-DMZ-TO-INSIDE rule 3 action drop
set firewall name FROM-DMZ-TO-INSIDE rule 3 protocol all
set firewall name FROM-DMZ-TO-INSIDE rule 3 source address 10.2.2.0/24
set firewall name FROM-DMZ-TO-INSIDE rule 10 action accept
set firewall name FROM-DMZ-TO-INSIDE rule 10 state established enable
set firewall name FROM-DMZ-TO-INSIDE rule 10 state related enable
commit
```

```
set firewall name FROM-OUTSIDE-TO-DMZ rule 3 action drop
```

```
set firewall name FROM-OUTSIDE-TO-DMZ rule 3 protocol all
set firewall name FROM-OUTSIDE-TO-DMZ rule 3 source address 192.1.1.0/24
set firewall name FROM-OUTSIDE-TO-DMZ rule 4 description "block this blacklisted attacker"
set firewall name FROM-OUTSIDE-TO-DMZ rule 4 action drop
set firewall name FROM-OUTSIDE-TO-DMZ rule 4 protocol all
set firewall name FROM-OUTSIDE-TO-DMZ rule 4 source address 195.142.44.18
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 description "outside access to ftp and smtps"
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 action accept
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 protocol tcp
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 source address !10.2.2.0/24
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 destination address 192.1.1.100
set firewall name FROM-OUTSIDE-TO-DMZ rule 10 destination port 20,587
set firewall name FROM-OUTSIDE-TO-DMZ rule 12 description "outside access to external dns"
set firewall name FROM-OUTSIDE-TO-DMZ rule 12 action accept
set firewall name FROM-OUTSIDE-TO-DMZ rule 12 protocol udp
set firewall name FROM-OUTSIDE-TO-DMZ rule 12 source address !10.2.2.0/24
set firewall name FROM-OUTSIDE-TO-DMZ rule 12 destination address 192.1.1.100
set firewall name FROM-OUTSIDE-TO-DMZ rule 12 destination port 53
```

```
set firewall name FROM-DMZ-TO-OUTSIDE rule 10 action accept
set firewall name FROM-DMZ-TO-OUTSIDE rule 10 state established enable
set firewall name FROM-DMZ-TO-OUTSIDE rule 10 state related enable
set firewall name FROM-DMZ-TO-OUTSIDE rule 10 source address 192.1.1.100
```

```
set zone-policy zone OUTSIDE from INSIDE firewall name FROM-INSIDE-TO-OUTSIDE
set zone-policy zone INSIDE from OUTSIDE firewall name FROM-OUTSIDE-TO-INSIDE
set zone-policy zone DMZ from INSIDE firewall name FROM-INSIDE-TO-DMZ
set zone-policy zone INSIDE from DMZ firewall name FROM-DMZ-TO-INSIDE
set zone-policy zone OUTSIDE from DMZ firewall name FROM-DMZ-TO-OUTSIDE
set zone-policy zone DMZ from OUTSIDE firewall name FROM-OUTSIDE-TO-DMZ
```