

[resources.intenseschool.com](http://resources.intenseschool.com)

## pfSense Series: Firewall Rules - Intense School


Welcome back to this series, in which we discuss and configure the various features of pfSense. In the previous article, we set up VLANs on pfSense so that we could use pfSense for inter-VLAN routing. In that article, we also touched a bit on firewall rules. In this article, we will take a deeper look at configuring firewall rules on pfSense.

### Firewall Rules

Among the most important features you will configure on a firewall are the firewall rules (obviously). When you install pfSense, all connections from the LAN are automatically permitted by default. However, all connections from the WAN are denied. We can view/configure firewall rules by navigating to **Firewall > Rules:**

Firewall: Rules

FloatingWANLAN

	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	
<input checked="" type="checkbox"/>		*	RFC 1918 networks	*	*	*	*	*		Block private networks	
<input checked="" type="checkbox"/>		*	Reserved/not assigned by IANA	*	*	*	*	*		Block bogon networks	
No rules are currently defined for this interface All incoming connections on this interface will be blocked until you add pass rules. Click the  button to add a new rule.											

FloatingWANLAN

	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	
<input checked="" type="checkbox"/>		*	*	*	LAN Address	80	*	*		Anti-Lockout Rule	
<input type="checkbox"/>		IPv4 *	LAN net	*	*	*	*	none		Default allow LAN to any rule	
<input type="checkbox"/>		IPv6 *	LAN net	*	*	*	*	none		Default allow LAN IPv6 to any rule	

☒ pass  
☐ pass (disabled)

☒ match  
☐ match (disabled)

☒ block  
☐ block (disabled)

☒ reject  
☐ reject (disabled)

☒ log  
☐ log (disabled)

Hint:

Rules are evaluated on a first-match basis (i.e. the action of the first rule to match a packet will be executed). This means that if you use block rules, you'll have to pay attention to the rule order. Everything that isn't explicitly passed is blocked by default.

**Hint:** In that article, we also saw that there are no firewall rules defined by default for new OPT interfaces. This means that any traffic seen on those interfaces will be denied, **even traffic destined to pfSense itself!**

Except for rules defined under the **Floating** tab, firewall rules process traffic in the inbound direction only, from top to bottom, and the process stops when a match is found. This is similar to how a Cisco router processes access lists, so one should be careful to put more specific rules at the top so that they are matched before generic rules.

Let's configure a sample security policy as follows:

- Any traffic from the LAN to any destination should be allowed.
- Allow ICMP from the DMZ to any destination.
- Allow SSH/HTTPS only from hosts 172.16.100.200 and 172.16.100.201 in the DMZ to the LAN network.
- Allow DNS, HTTP, and HTTPS from the DMZ to the Internet.
- Deny everything else!

**Note:** Because I'm trunking the VMware interface used for both LAN and DMZ, I may not be able to access the webGUI from the host PC anymore via the LAN IP address. Therefore, I will leave the rule for WAN access open. Keep in mind that, if you are using DHCP, the host PC's IP address may change from the one you configured in the firewall rule and you won't be able to access the webGUI anymore (depending on how strict your rule was).

#### ***Policy #1: Permit all traffic from LAN***

As we have seen above, all traffic (IPv4 and IPv6) from the LAN is permitted by default. Therefore, we don't need to do anything extra to configure this security policy.

#### ***Policy #2: Permit ICMP from DMZ***

In the last article, we configured a firewall rule that allows ICMP from the DMZ to any destination, as shown below:



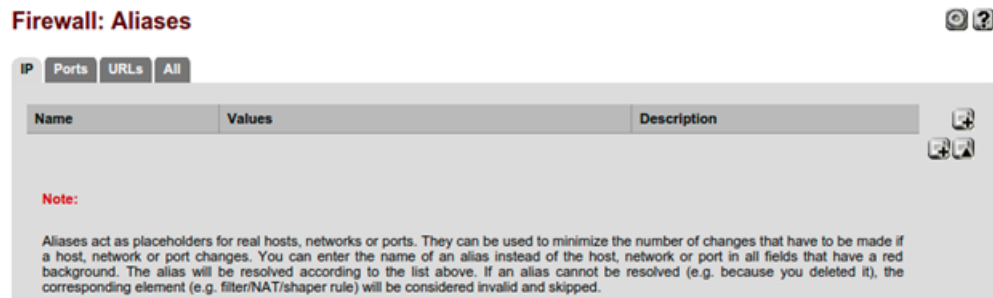
Floating										
WAN LAN DMZ										
ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	
IPv4 ICMP		*	*	*	*	*	none		Allow ICMP any any	

Let's leave this rule configured but, by walking through the steps of configuring firewall rules for policy #3 and #4, you can understand how this rule was configured.

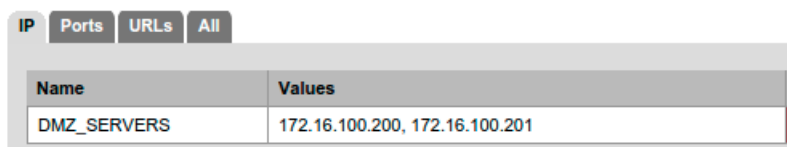
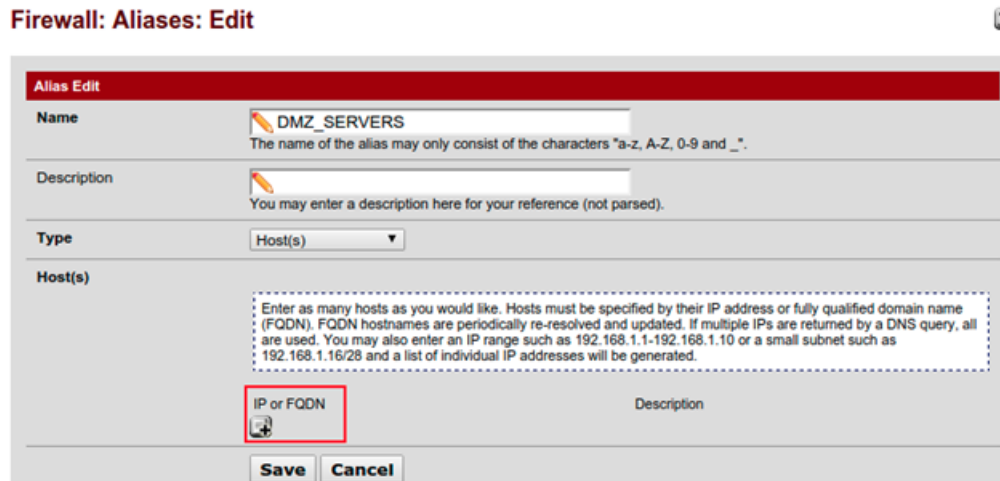
***Policy #3: Permit SSH/HTTPS from 172.16.100.200 and 172.16.100.201 to LAN***

I decided to include this policy here so that we could see another feature available in pfSense – Aliases. This feature is similar to object groups on the Cisco IOS, where we group similar objects together to make configuration simpler. With aliases, instead of specifying the individual objects, you just specify the alias name.

Therefore, let's configure two aliases: one for SSH and HTTPS and the second one for the hosts 172.16.100.200 and 172.16.100.201. To do this, we will navigate to **Firewall > Aliases**:



As you can see, we can create aliases for IP, Ports, and URLs. We will start with the one for IP and then move to the one for ports.



IP

Ports

URLs

All

Name	Values
SSH_HTTPS	22, 443

When you are done with your configuration, apply your changes and we can move on to creating the firewall rule itself. We will navigate to **Firewall > Rules** and then select the **DMZ** tab. The settings for my own rule are shown below:

### Firewall: Rules: Edit

**Edit Firewall rule**

<b>Action</b>	Pass <small>Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.</small>
<b>Disabled</b>	<input type="checkbox"/> <b>Disable this rule</b> <small>Set this option to disable this rule without removing it from the list.</small>
<b>Interface</b>	DMZ <small>Choose which interface packets must be sourced on to match this rule.</small>
<b>TCP/IP Version</b>	IPv4 <small>Select the Internet Protocol version this rule applies to</small>
<b>Protocol</b>	TCP <small>Choose which IP protocol this rule should match. Hint: in most cases, you should specify TCP here.</small>
<b>Source</b>	<input type="checkbox"/> <b>not</b> <small>Use this option to invert the sense of the match.</small> Type: Single host or alias Address: DMZ_SERVERS / Advanced - Show source port range
<b>Destination</b>	<input type="checkbox"/> <b>not</b> <small>Use this option to invert the sense of the match.</small> Type: LAN net Address: /
<b>Destination port range</b>	from: (other) SSH_HTT to: (other) SSH_HTT <small>Specify the port or port range for the destination of the packet for this rule. Hint: you can leave the 'to' field empty if you only want to filter a single port</small>
<b>Log</b>	<input type="checkbox"/> <b>Log packets that are handled by this rule</b> <small>Hint: the firewall has limited local log space. Don't turn on logging for everything. If you want to do a lot of logging, consider using a remote syslog server (see the <a href="#">Diagnostics: System logs: Settings</a> page).</small>
<b>Description</b>	Allow SSH/HTTPS from DMZ hosts to LAN <small>You may enter a description here for your reference.</small>

As you may have noticed when creating the port aliases, you don't specify the protocol. It is when we are creating the firewall rule that we specify the protocol, as shown above. Also notice how we specified the source as the alias we created—once you start typing the name, aliases that match that name show up. We also used the alias we created for the ports under the *Destination port range* field. Finally, there are some default names such as *LAN address* (i.e., LAN interface IP address of pfSense) and *LAN net* (i.e., LAN network and other static routes configured on that interface) that we can use when configuring rules. These make your life easier because, if an address/network changes, you won't have to alter the rule as the rule will be automatically updated to match the new address(es).

***Policy #4: Allow DNS, HTTP, and HTTPS from DMZ to Internet***

There are several ways you can configure this rule, depending on how restrictive you want your rule to be. DNS (not zone transfers) uses UDP port 53 by default, while HTTP and HTTPS use TCP port 80 and 443, respectively. If you create a port alias matching the three protocols, you will have to use “TCP/UDP” in the Protocol field of the firewall rule. This means that TCP/UDP ports 53, 80 and 443 will be allowed which is more than you want.

Let's practice the principle of least privilege and be as restrictive as possible. We will create a port alias for HTTP and HTTPS and then create a standalone rule for DNS.

	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
<input type="checkbox"/>	1	IPv4 ICMP	*	*	*	*	*	none		Allow ICMP any any
<input type="checkbox"/>	2	IPv4 TCP	DMZ_SERVERS	*	LAN net	SSH HTTPS	*	none		Allow SSH/HTTPS from DMZ hosts to LAN
<input type="checkbox"/>	3	IPv4 TCP	DMZ net	*	*	HTTP S	*	none		Allow HTTP/S from DMZ to Internet
<input type="checkbox"/>	4	IPv4 UDP	DMZ net	*	*	53 (DNS)	*	none		Allow DNS from DMZ to Internet

If you were able to identify a gap in this our configuration, I salute your observation skills. Because firewall rules apply to traffic coming into an interface and since we didn't specify a destination network, it means this last rule we just created also allows hosts on the DMZ to open DNS, HTTP, and HTTPS connections to the LAN!

To remedy this situation, we need to add a rule that blocks traffic from the DMZ network to the LAN and place this rule between Policy #3 and Policy #4.

First, let's create the rule: by default, new rules are added at the bottom.

	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
<input type="checkbox"/>	1	IPv4 ICMP	*	*	*	*	*	none		Allow ICMP any any
<input type="checkbox"/>	2	IPv4 TCP	DMZ_SERVERS	*	LAN net	SSH HTTPS	*	none		Allow SSH/HTTPS from DMZ hosts to LAN
<input type="checkbox"/>	3	IPv4 TCP	DMZ net	*	*	HTTP S	*	none		Allow HTTP/S from DMZ to Internet
<input type="checkbox"/>	4	IPv4 UDP	DMZ net	*	*	53 (DNS)	*	none		Allow DNS from DMZ to Internet
<input checked="" type="checkbox"/>	5	IPv4 *	*	*	LAN net	*	*	none		Block all traffic from DMZ to LAN

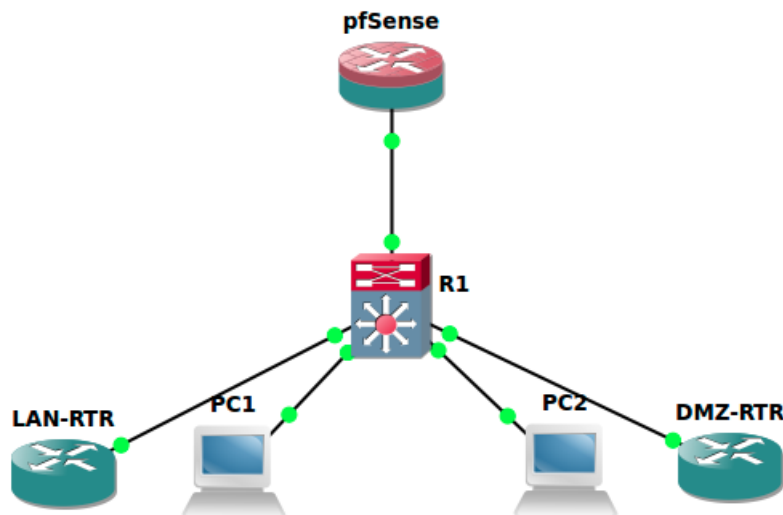
To move the rule to the correct position, we will select the checkbox in front of the rule and click the “Move

selected rules before this rule” button for the rule which we want the selected rules to precede (highlighted above):

Floating WAN LAN DMZ										
	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
<input type="checkbox"/>	▶	IPv4	ICMP	*	*	*	*	none		Allow ICMP any any
<input type="checkbox"/>	▶	IPv4	TCP	DMZ_SERVERS	*	LAN net	SSH HTTPS	*	none	Allow SSH/HTTPS from DMZ hosts to LAN
<input type="checkbox"/>	✖	IPv4	*	*	LAN net	*	*	none		Block all traffic from DMZ to LAN
<input type="checkbox"/>	▶	IPv4	TCP	DMZ net	*	*	HTTP S	*	none	Allow HTTP/S from DMZ to Internet
<input type="checkbox"/>	▶	IPv4	UDP	DMZ net	*	*	53 (DNS)	*	none	Allow DNS from DMZ to Internet

With this, we have come to the end of our rules definition. The last policy says that everything else should be denied, but that is already implicit in the rules table (just like a Cisco ACL). Explicitly defining a “deny all” rule is useful when you want to log such traffic.

It is always advisable to test your firewall rules to make sure you have not accidentally permitted traffic that should be blocked or denied traffic that should be allowed. In our case, we may want to add some smarter devices (than VPCS) onto the LAN and DMZ that will allow us open SSH and HTTPS connections. Therefore, our GNS3 topology now looks like this:



**Note:** I have basic IP configuration on the routers. I have also enabled SSH on the LAN-RTR. Both routers are configured to use pfSense as their DNS server.

Let’s begin our test by checking that the LAN-RTR can ping an Internet URL (i.e., DNS and ICMP):

```

LAN-RTR#ping google.com
Translating "google.com"...domain server (172.16.215.100) [OK]
  
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 197.253.18.123, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 46/67/81 ms
```

Next we will ping from a DMZ host to the LAN since ICMP from the DMZ is allowed to any destination (policy #2):

```
DMZ-RTR#ping 172.16.215.201
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.215.201, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/26/42 ms
```

To test the third policy, I will open an SSH connection from the DMZ-RTR to the LAN-RTR:

```
DMZ-RTR#ssh -l cisco 172.16.215.201
Password:

LAN-RTR>
```

For the fourth policy, I can ping from the DMZ-RTR to an Internet URL. Since this will involve DNS, we can confirm that our fourth policy works:

```
DMZ-RTR#ping google.com
Translating "google.com"...domain server (172.16.100.1) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 197.253.18.123, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 69/74/83 ms
```

Just to confirm that our deny rule works (the one denying DMZ from accessing the LAN), I will change the IP address of the DMZ-RTR from 172.16.100.201 to 172.16.100.220 and try to open SSH to LAN-RTR again. As shown below, it won't work:

```
DMZ-RTR(config)#interface e0/0
DMZ-RTR(config-if)#ip addr 172.16.100.220 255.255.255.0
DMZ-RTR(config-if)#
DMZ-RTR(config-if)#do ssh -l cisco 172.16.215.201
DMZ-RTR(config-if)#
```

Although the webGUI doesn't (yet) provide a way to check the counters on firewall rules, we can use the following command through the Shell: **pfctl -vvsr**:

```
@89(1456227294) block drop in quick on le0_vlan20 inet from any to
172.16.215.0/24 label "USER_RULE: Block all traffic from DMZ to LAN"
```

```
[ Evaluations: 11          Packets: 6          Bytes: 264          States: 0
]
```

```
[ Inserted: pid 25519 State Creations: 18446735277677785112]
```

1. @89(1456227294) block drop in quick on le0\_vlan20 inet from any to 172.16.215.0/24 label "USER\_RULE: Block all traffic from DMZ to LAN"

- 2.

3. [ Evaluations: 11 Packets: 6 Bytes: 264 States: 0 ]

- 4.

5. [ Inserted: pid 25519 State Creations: 18446735277677785112]

@89(1456227294) block drop in quick on le0\_vlan20 inet from any to 172.16.215.0/24 label "USER\_RULE: Block all traffic from DMZ to LAN"

[ Evaluations: 11                      Packets: 6                      Bytes: 264                      States: 0  
]

[ Inserted: pid 25519 State Creations: 18446735277677785112]

**Note:** To access the Shell, enter option 8 at the console of pfSense or via the terminal when connected via SSH.

```
*** Welcome to pfSense 2.2.6-RELEASE-pfSense (amd64) on pfSense ***

WAN (wan)      -> em0      -> v4/DHCP4: 192.168.8.101/24
LAN (lan)      -> le0_vlan10 -> v4: 172.16.215.100/24
DMZ (opt1)     -> le0_vlan20 -> v4: 172.16.100.1/24

0) Logout (SSH only)      9) pfTop
1) Assign Interfaces      10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) pfSense Developer Shell
4) Reset to factory defaults 13) Upgrade from console
5) Reboot system          14) Disable Secure Shell (sshd)
6) Halt system            15) Restore recent configuration
7) Ping host              16) Restart PHP-FPM
8) Shell

Enter an option: █
```

## Summary

This brings us to the end of this article, in which we have configured firewall rules on pfSense. I hope you have found this article insightful and I look forward to writing the next one in the series.

## References

-



- 
- Firewall rule processing order: [https://doc.pfsense.org/index.php/Firewall\\_Rule\\_Processing\\_Order](https://doc.pfsense.org/index.php/Firewall_Rule_Processing_Order)