OPNsense Firewall

- Sources and inspiration
  - Set Up a Fully Functioning Home Network Using OPNsense
  - Videos
    - Set up a Full Network using OPNsense (Part 1: Overview)
    - Set up a Full Network using OPNsense (Part 2: OPNsense)
    - Set up a Full Network using OPNsense (Part 3: Switch)
    - Set up a Full Network using OPNsense (Part 4: Wireless Access Point))

Firewall Configuration

The time has come to create firewall rules. Firewall rules are critical for providing increased security among the devices in your network. Having a solid understanding in this area will be crucial in helping you lock down your network tighter.

As you likely know, no software or hardware is fully impenetrable, which is why it is important to have several layers of defense when protecting your network.

I decided to include several VLANs in this guide in order to demonstrate different types of networks you may want to implement in your home network. They should provide some good use cases for how the firewall rules may look different for the types of access desired. Of course, you do not need to implement all of these VLANs in your network.

Note

I will be making use of firewall aliases in the firewall rules below, so if you see a name instead of an IP address for the “Source” or “Destination” it means I am using either a built-in firewall alias or the custom firewall aliases I describe in the “Firewall: Aliases” section.

The names in the firewall rules are not hostnames of devices on the network because hostnames can only be used in firewall aliases. Firewall rules only allow you to enter a single IP/network address or a single firewall alias (aliases may contain more than one value). If you want to use multiple IP addresses or network addresses in a single firewall rule, you have to create an alias containing those addresses and use that alias in the firewall rule.

Firewall > Aliases

Create a new firewall alias by visiting the “Firewall > Aliases” page and clicking the “+” button at the bottom of the page. For IPv4 addresses you could create the following alias:

	PrivateNetworks	**HomeAssistantServer**	**HPPrinter**		**AdminPCs**
Enabled	**Checked**	**Checked**	**Checked**	**Checked**	**Checked**
Name	**PrivateNetworks**	**HomeAssistantServer**	**HPPrinter**	**HPE1820**	**AdminPCs**
Type	**Network(s)**	**Host(s)**	**Host(s)**	**Host(s)**	**Host(s)**
Categories					
Content	**__lan_network, __opt1_network, __opt2_network,__opt3_network**	**192.168.120.8**	**192.168.110.8**	**192.168.101.2**	**192.168.110.16-192.168.110.24**
Statistics	**Unchecked**	**Unchecked**	**Unchecked**	**Unchecked**	
Description	**All my local networks**	**Home Assistant Server for Train software**	**HP Printer**	**PC that can manage network item**	

Click the ***“Apply”*** button to ensure the alias changes are applied. If you do not click “Apply”, the new aliases will not be available to select when creating firewall rules.

Firewall > NAT > Port Forward

!!!! These rules are only for testing at home, from my workplace, so it is easier to create documentation.
!!! Important they must be disabled or deleted during installation.

Server	**OPNsense**	**HPE 1820**	**Home Assistant**
Disabled	**Checked**	**Checked**	**Checked**
No RDR (NOT)	Unchecked	Unchecked	Unchecked
Interface	**WAN**	**WAN**	**WAN**
TCP/IP Version	**IPv4**	**IPv4**	**IPv4**
Protocol	**TCP**	**TCP**	**TCP**
Destination / Invert	Unchecked	Unchecked	Unchecked
Destination	**WAN address**	**WAN address**	**WAN address**
Destination port range from:	**HTTP**	**(other) 8080**	**(other) 8123**
Destination port range to:	**HTTP**	**(other) 8080**	**(other) 8123**
Redirect target IP	**Single host or Network**	**Single host or Network**	**Single host or Network**
	192.168.101.1	**192.168.101.2**	**192.168.120.8**
Redirect target port	**HTTP**	**HTTP**	**8123**
Pool Options:	Default	Default	Default
Log	Unchecked	Unchecked	Unchecked
Category			
Description	**Nat to OPNsense**	**Nat to HPE 1820**	**Nat to Home Assistant**
Set local tag			
Match local tag			
No XMLRPC Sync	Unchecked	Unchecked	Unchecked
NAT reflection	Use system default	Use system default	Use system default
Filter rule association	**Pass**	**Pass**	**Pass**

Firewall > Rules > WAN

Only Automatically generated rules here

Firewall > Rules > LAN

The LAN network will already have the ***"allow all IPv4"*** rules created by default from the OPNsense installation. However, I will tweak them so that access to your other networks is limited. If you recall from earlier, in this example I am using the ***untagged LAN*** as the ***management network*** where all of the ***critical network*** infrastructure will be ***managed***.

Option	**DNS**	**NTP Time**	**ICMP**	**HomeAssistant**	**HPPrinter**	**Access only to Internet**
Action	**Pass**	**Pass**	**Pass**	**Pass**	**Pass**	**Pass**
TCP/IP Version	**IPv4**	**IPv4**	**IPv4**	**IPv4**	**IPv4**	**IPv4**
Protocol	**TCP/UDP**	**UDP**	**ICMP**	**TCP**	**TCP/UDP**	**any**

Option	DNS	NTP Time	ICMP	HomeAssistant	HPPrinter	Access only to Internet
Source	**LAN net**	**LAN net**	**LAN net**	**AdminPCs**	**AdminPCs**	
Destination / Invert	**Unchecked**	**Unchecked**	**Unchecked**	**Unchecked**	**Unchecked**	**Checked**
Destination	**LAN address**	**LAN address**	**any**	**HomeAssistantServer**	**HPPrinter**	**PrivateNetworks**
Destination port range from:	**DNS**	**NTP**	**any**	**(other) 8123**	**any**	**any**
Destination port range to:	**DNS**	**NTP**	**any**	**(other) 8123**	**any**	**any**
Description	**Allow access to DNS**	**Allow access to NTP**	**Allow ICMPv4 from LAN to all network**	**Allow access for Admins to Home Assistant server**	**Allow access to Printer for Admins PC**	**Allow access only to Internet**

Firewall > Rules > Office

The Office network can be used for PCs, laptops, or phones. The primary purpose is to separate these devices from potentially more vulnerable IOT devices. The rules will be similar to what has been shown until this point except for rules to allow access to devices/services located on other networks.

For the allow rule for the printer, I set the destination port to any to keep it simple but printers can often require several ports depending on the type of printer. For instance, HP printers have a list of ports that you may add in order to further restrict access.

Option	DNS	NTP Time	HomeAssistant	HPE1820	Access only to Internet
Action	**Pass**	**Pass**	**Pass**	**Pass**	**Pass**
TCP/IP Version	**IPv4**	**IPv4**	**IPv4**	**IPv4**	**IPv4**
Protocol	**TCP/UDP**	**UDP**	**TCP**	**TCP**	**any**
Source	**Office net**	**Office net**	**AdminPCs**	**AdminPCs**	**Office net**
Destination / Invert	**Unchecked**	**Unchecked**	**Unchecked**	**Unchecked**	**Checked**
Destination	**Office address**	**Office address**	**HomeAssistantServer**	**HPE1820**	**PrivateNetworks**
Destination port range from:	**DNS**	**NTP**	**(other) 8123**	**HTTP**	**any**
Destination port range to:	**DNS**	**NTP**	**(other) 8123**	**HTTP**	**any**
Description	**Allow access to DNS**	**Allow access to NTP**	**Admin Accesss to Home Assistant Server**	**Admin Accesss to HPE 1820**	**Allow access only to Internet**

Ideally, you should have a dedicated device (or VM perhaps) residing on the LAN that has access to all of the web interfaces of your network infrastructure. If you do not have any dedicated devices available, you could create a rule on the USER network to allow your PC/laptop to access the management interfaces. This is less than ideal because you are poking a hole in the management network, but at least access is still restricted from a specific device to specific web interfaces and ports. Security is still better than a flat network with full access to everything.

Firewall > Rules > Train

The Train network is where you can put less trusted IOT (Internet of Things) type devices or perhaps devices that no longer receive security updates – anything that is likely more vulnerable than your other devices which are regularly updated (PCs, laptops, etc). The reason for such a network is that IOT devices have been notoriously prone to vulnerabilities. They are often developed as cheap, convenient devices which do not receive thorough security audits and are not updated as frequently (or at all if users do not perform updates).

Option	**DNS**	**NTP Time**	**Home Assistatnt**
Action	**Pass**	**Pass**	**Pass**
TCP/IP Version	**IPv4**	**IPv4**	**IPv4**
Protocol	**TCP/UDP**	**UDP**	**any**
Source	**Train net**	**Train net**	**HomeAssistantServer**
Destination / Invert	**Unchecked**	**Unchecked**	**Checked**
Destination	**Train address**	**Train address**	**PrivateNetworks**
Destination port range from:	**DNS**	**NTP**	**any**
Destination port range to:	**DNS**	**NTP**	**any**
Description	**Allow access to DNS**	**Allow access to NTP**	**Allow HA Server access only to Internet**

Firewall > Rules > IPCam

The IPCAM network is where you can put IP security cameras that is isolated from your network and the Internet. This can be useful if you have a NVR box where you access all your camera feeds. In this network, you could simply allow access to DNS and everything else will be blocked.

You do not really need to create any rules at all to block all access (access is “deny all” by default when no rules exist), but I found that if you allow DNS, the firewall logs are a lot cleaner because you do not see all of the DNS blocks. IP cameras often try to phone home for cloud services so if DNS is blocked they may try even harder to query DNS, which increases spam to your firewall logs. Since all other access is blocked, even if the IP addresses are resolved for domain names, access to external servers is still blocked.