LOGZILLA DOCUMENTATION

Netgate pfSense

Rules, dashboards, and triggers for Netgate pfSense firewall (filterlog) and DHCP (dhcpd) syslog

LogZilla App Store · Generated June 17, 2026 · logzilla.ai/docs/logzilla-appstore/pfsense

Overview

Netgate pfSense is an open-source firewall and router platform built on FreeBSD. pfSense forwards two high-value syslog streams: `filterlog` (packet filter block/pass decisions) and `dhcpd` (DHCP lease lifecycle). The pfSense app parses both streams into LogZilla user tags for searching, dashboards, and aggregate reporting.

App Function

- Parses `filterlog` CSV records for IPv4 and IPv6 traffic across all protocols (TCP, UDP, ICMP, ICMPv6, CARP, IGMP, and others).
- Extracts firewall metadata: action (pass/block), direction, matched rule (pf tracker ID), interface, protocol, source/destination IP, and destination port.
- Parses `dhcpd` lease events (DISCOVER, OFFER, REQUEST, ACK, NAK) for client MAC, leased IP, client hostname, and interface.
- Converts numeric ports to service names (`443` to `https`) and numeric IP protocols to names (`6` to `TCP`, `112` to `VRRP`).
- Classifies events with the cross-vendor taxonomy: firewall traffic as Network / Access Control, DHCP as Network / Lease, and auto-assigns the applicable compliance frameworks.
- Flags blocked inbound connection attempts to exposed management ports (SSH, RDP, SMB, Telnet, VNC, NetBIOS) as Security / Threat and maps them to MITRE ATT&CK T1046 (Network Service Discovery).
- Ships with Firewall and DHCP dashboards plus a trigger for blocked inbound traffic.

Vendor Documentation

- pfSense Documentation (https://docs.netgate.com/pfsense/en/latest/)
- Raw Filter Log Format (https://docs.netgate.com/pfsense/en/latest/monitoring/logs/raw-filter-format.html)
- Viewing the Firewall Log (https://docs.netgate.com/pfsense/en/latest/monitoring/logs/firewall.html)
- Remote Logging with Syslog (https://docs.netgate.com/pfsense/en/latest/monitoring/logs/settings.html)

LogZilla Configuration

pfSense requires a dedicated syslog port in LogZilla. The `dhcpd` program name collides with ISC DHCP on other platforms, so a dedicated port keeps pfSense traffic isolated.

Navigate to **Settings > System > Application Ports**.
Set **Netgate pfSense syslog port** to a dedicated port (e.g. `5526`).
Click **Save**.
The syslog and parser services reload automatically. Both TCP and UDP listeners are enabled on the configured port.

LogZilla Cloud: the Application Ports page is not available on the cloud platform, where devices reach LogZilla through a relay rather than a direct syslog listener. To enable parsing there, tag pfSense events at the relay with `_source_type=pfsense`. See Application Ports on LogZilla Cloud (https://www.logzilla.ai/docs/logzilla-cloud/application-ports-on-cloud).

Device Configuration

Configure pfSense to forward logs to the LogZilla server:

Navigate to **Status > System Logs > Settings**.
Enable **Send log messages to remote syslog server**.
Set **Remote log servers** to `LOGZILLA_SERVER:5526`.
Under **Remote Syslog Contents**, enable **Firewall Events** and **DHCP service events** (or **Everything**).
Click **Save**.

Verification

After saving, generate firewall or DHCP activity on the pfSense device, then confirm the events arrive in LogZilla. In the LogZilla UI, search for events from the device and confirm the **Vendor** tag shows `Netgate` and **Product** shows `pfSense`. If no events appear, confirm the remote syslog server port matches the **Syslog Pfsense Port** value under **Settings -> System Settings -> Application Ports**, and that the network path from pfSense to the LogZilla server permits the configured port.

Incoming Log Format

pfSense can forward syslog in either the default BSD (RFC 3164) format or the optional RFC 5424 format (set on the pfSense device under Status -> System Logs -> Settings, "Log Message Format"). Both carry the program name (`filterlog` or `dhcpd`), so the parser works with either; RFC 5424 is recommended because the BSD format omits the sending hostname. The parser reads the message body that follows the syslog header.

`filterlog` (CSV), common fields followed by an IP-version and protocol-specific tail:

```
rulenum,subrule,anchor,tracker,interface,reason,action,direction,ipversion,...
```

`dhcpd` (free text):

```
DHCPACK on <ip> to <mac> (<hostname>) via <interface>
```

Parsed Metadata Fields

Tag Name	Example	Description
`Vendor`	`Netgate`	Vendor name
`Product`	`pfSense`	Product name
`Event Class`	`Network`, `Security`	Cross-vendor classification (`Security` for recon hits)
`Event Type`	`Access Control`, `Lease`, `Threat`	Event subtype (`Access Control` for filterlog, `Lease` for dhcpd, `Threat` for recon)
`MitreId`	`T1046`	MITRE ATT&CK technique ID (blocked inbound hits to management ports)
`MITRE Tactic`	`Discovery`	MITRE ATT&CK tactic (paired with the MitreId)
`Action`	`block`	Firewall action, `pass` or `block` (filterlog)
`Direction`	`in`	Traffic direction, `in` or `out` (filterlog)
`Protocol`	`TCP`	Protocol name: TCP, UDP, ICMP, ICMPv6, VRRP (filterlog)
`Interface`	`vtnet1.10`	pf interface (filterlog and dhcpd)
`SrcIP`	`198.51.100.7`	Source IP address (filterlog) (HC)
`DstIP`	`10.0.0.5`	Destination or leased IP address (filterlog and dhcpd) (HC)
`DstPort`	`https`	Destination port, as a service name (filterlog TCP/UDP)
`SrcMAC`	`00:00:5e:00:53:01`	DHCP client MAC address (dhcpd) (HC)
`Hostname`	`client1`	DHCP client hostname (dhcpd) (HC)
`pfSense Rule ID`	`1000017361`	pf rule tracker ID that matched (filterlog)

MITRE ATT&CK Mapping

pfSense filterlog records carry no threat classification of their own, so the parser infers a single, conservative technique from packet behavior rather than from a vendor field: a blocked **inbound** connection attempt to an exposed management service port is treated as reconnaissance.

Condition	Technique	Tactic
`Action=block`, `Direction=in`, `DstPort` in ssh, telnet, netbios-ssn, microsoft-ds, ms-wbt-server, vnc	`T1046` Network Service Discovery	Discovery

Matching events are reclassified from Network / Access Control to Security / Threat. All other firewall and DHCP events carry no MITRE tag.

Log Examples

Firewall, IPv4 TCP pass:

```
148,,,1000017361,vtnet0,match,pass,out,4,0x0,,63,0,0,DF,6,tcp,64,10.0.0.5,192.0.2.10,22438,443,0,SEC,
893929139,,65535,,mss;nop;wscale;nop;nop;TS;sackOK;eol
```

Firewall, IPv4 UDP block:

```
4,,,1000000103,vtnet1.20,match,block,in,4,0x0,,128,17302,0,none,17,udp,78,10.1.1.10,10.1.1.255,53117,
53,58
```

Firewall, blocked inbound SSH attempt (recon, mapped to T1046):

```
4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,12345,0,DF,6,tcp,60,198.51.100.7,10.0.0.5,51000,22,0,S
,1000000001,,64240,,mss;sackOK;TS;nop;wscale
```

Firewall, IPv6 UDP block:

```
6,,,1000000105,vtnet1.10,match,block,in,6,0x00,0xb0900,255,UDP,17,168,fe80::2,ff02::fb,5353,5353,168
```

DHCP lease acknowledgement:

```
DHCPACK on 10.1.1.50 to 00:00:5e:00:53:01 (client1) via vtnet1.10
```

DHCP pool exhaustion:

```
DHCPDISCOVER from 00:00:5e:00:53:02 via vtnet1.20: network 10.1.1.0/24: no free leases
```

Dashboards

- **pfSense: Firewall** - the security and traffic view: block/pass rates, top blocked sources and destination ports, top matching rules, protocol mix, and a live event stream.
- **pfSense: DHCP** - the DHCP service view: lease volume, unique clients and leased IPs, top client hostnames and MACs, leases per interface, and a live event stream.

Triggers

- **pfSense: Management Port Recon** - fires on the reconnaissance signal (`Event Type` `Threat`, `MitreId` `T1046`): a blocked inbound attempt to an exposed management port. Narrow and high-signal.
- **pfSense: Inbound Traffic Blocked** - fires on firewall events where `Action` is `block` and `Direction` is `in`. Because block logs can be high volume, narrow the filter (add a specific `DstPort` or `Interface`) to match the alerting policy for a given deployment.