

Security event analysis using Wazuh

Roozah Khan

As a Security Information and Event Management (SIEM) system, Wazuh's capability is not limited to host-based agent monitoring. You can also let other applications, such as pfSense, to forward all security logs to Wazuh, and analyze them all in the same interface.

In the last lab, we will configure pfSense to forward its firewall log to Wazuh and visualize it.

1. Config pfSense to send remote log.

As a firewall, pfSense is able to store various logs related to the filtering events. Such logs can either be stored locally, or, more commonly, be sent to a remote server that stores and analyzes log files. To enable this, log into pfSense, and select "Status → System Logs" menu.

You will be brought to system log configuration page. Select "Settings" tab, scroll all the way down till you see the "Remote Logging Options" section.

Check "send log messages to remote syslog server".

Chose IPv4 protocol.

Set remote log server IP to be 7.7.7.2, that is, the Wazuh manager. Leave the port empty, which will use the default port UDP/514.

Check "everything" in "remote syslog contents" to send all logs to the Wazuh manager.

Save the settings.

Provide a screenshot showing your configuration of remote log in pfsense. – screenshot 1.

Remote Logging Options

Enable Remote Logging

☒ Send log messages to remote syslog server

Source Address

Default (any)

This option will allow the logging daemon to bind to a single IP address, rather than all IP addresses. If a single IP is picked, remote syslog servers must all be of that IP type. To mix IPv4 and IPv6 remote syslog servers, bind to all interfaces.

NOTE: If an IP address cannot be located on the chosen interface, the daemon will bind to all addresses.

IP Protocol

IPv4

This option is only used when a non-default address is chosen as the source above. This option only expresses a preference; If an IP address of the selected type is not found on the chosen interface, the other type will be tried.

Remote log servers

7.7.7.2

IP[port]

IP[port]

Remote Syslog Contents

☒ Everything

☐ System Events
☐ Firewall Events
☐ DNS Events (Resolver/unbound, Forwarder/dnsmasq, filterdns)
☐ DHCP Events (DHCP Daemon, DHCP Relay, DHCP Client)
☐ PPP Events (PPPoE WAN Client, L2TP WAN Client, PPTP WAN Client)
☐ General Authentication Events
☐ Captive Portal Events
☐ VPN Events (IPsec, OpenVPN, L2TP, PPPoE Server)
☐ Gateway Monitor Events
☐ Routing Daemon Events (RADVD, UPnP, RIP, OSPF, BGP)
☐ Network Time Protocol Events (NTP Daemon, NTP Client)
☐ Wireless Events (hostapd)

Syslog sends UDP datagrams to port 514 on the specified remote syslog server, unless another port is specified. Be sure to set syslogd on the remote server to accept syslog messages from pfSense.

Save

2. Config Wazuh to accept remote log.

Now that the pfSense is sending log files to Wazuh, we next need to config Wazuh to accept it.

From Wazuh manager, open `/var/ossec/etc/ossec.conf`.

Find the section `<remote>` and make sure it has the following configuration.

```
<remote>

  <connection>syslog</connection>

  <port>514</port>

  <protocol>udp</protocol>

  <allowed-ips>7.7.7.0/28</allowed-ips>

</remote>
```

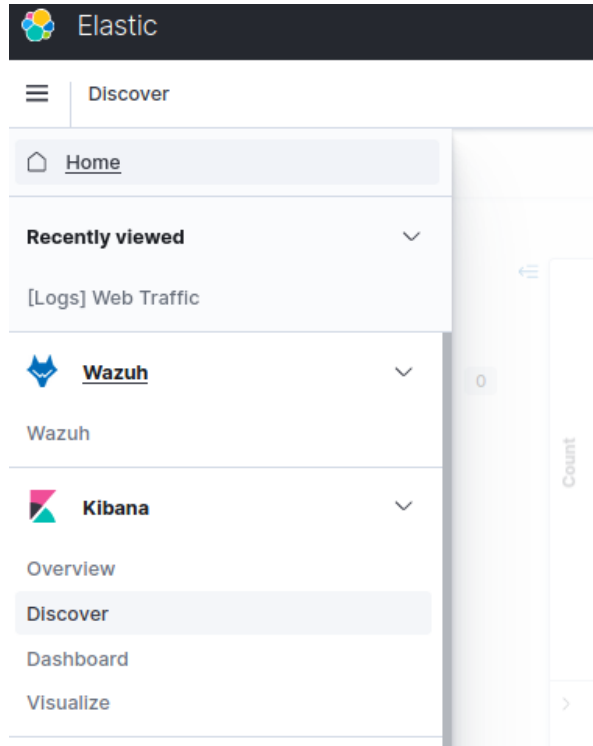
After the modification, save the file, and restart wazhu manager by `systemctl restart wazuh-manager`.

This finishes the configuration on both pfsense and wazuh.

3. Verify the remote log is working.

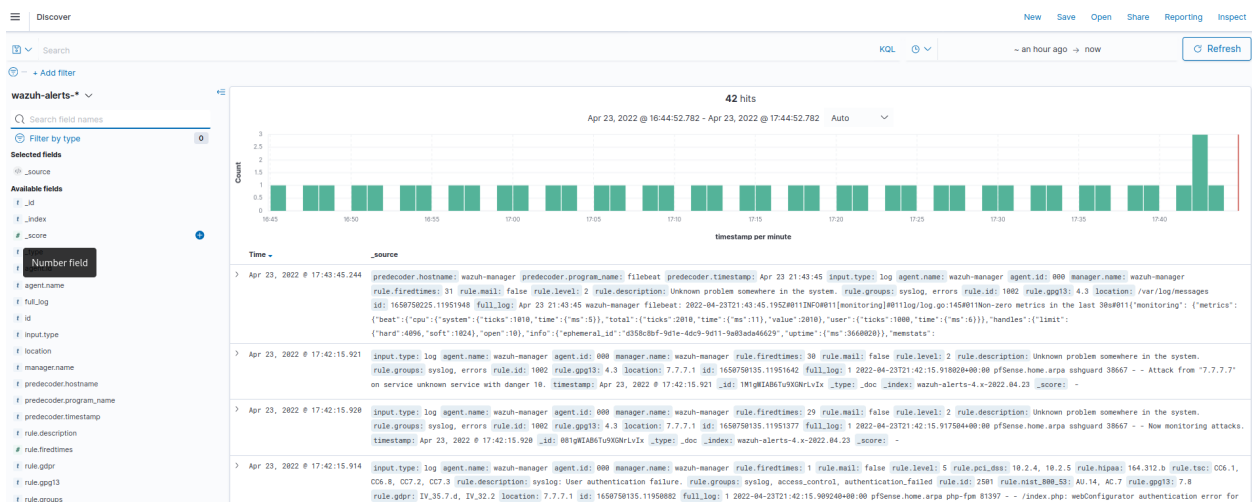
Log out of the pfSense graphical interface. Re-login with an incorrect password. And then login with the correct password. This will trigger an “authentication failure” event on pfSense, and such event should be forwarded to wazuh.

In Wazuh, select “Kibana → discover”, as shown in the figure.



On the top, there is a field to set the interval of events to display. Change it to a reasonable time interval, such as the last hour.

Click “refresh”, you should see many events are displayed, similar to the following figure.



Do a search, with the keyword “authentication”, you should see at least one event with key phases “authentication error ...”.

Inspect the event, it shows many details regarding when and where this event happened.

Provide a screenshot showing that you have successfully found this event. Zoom your web browser to make the font larger, and your screenshot should focus on the event and the content should be recognizable. – screenshot 2.

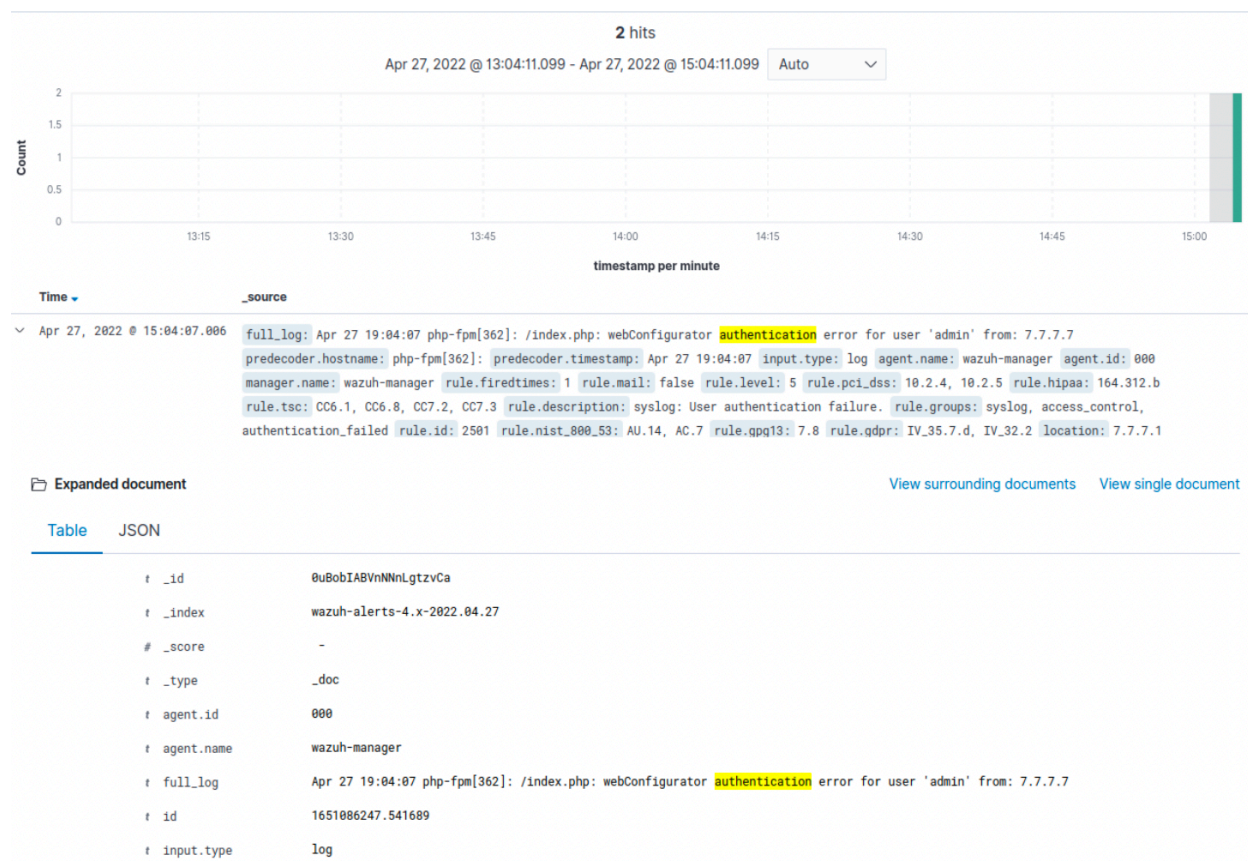
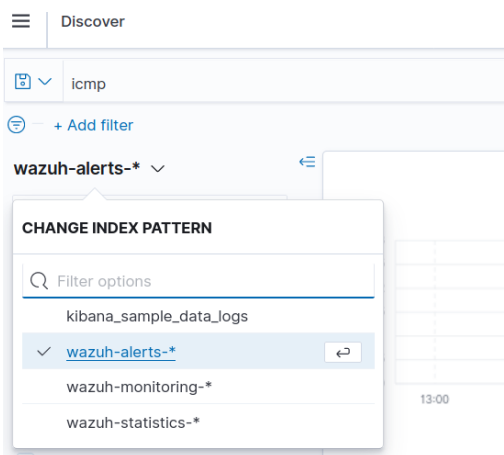


Table	JSON
t _id	0uBobIABVnNnLgtzvCa
t _index	wazuh-alerts-4.x-2022.04.27
# _score	-
t _type	_doc
t agent.id	000
t agent.name	wazuh-manager
t full_log	Apr 27 19:04:07 php-fpm[362]: /index.php: webConfigurator authentication error for user 'admin' from: 7.7.7.7
t id	1651006247.541689
t input.type	log
t location	7.7.7.1
t manager.name	wazuh-manager
t predecoder.hostname	php-fpm[362]:
t predecoder.timestamp	Apr 27 19:04:07
t rule.description	syslog: User authentication failure.
# rule.firedtimes	1
t rule.gdpr	IV_35.7.d, IV_32.2
t rule.gpg13	7.8
t rule.groups	syslog, access_control, authentication_failed
t rule.hipaa	164.312.b
t rule.id	2501
# rule.level	5
rule.mail	false
t rule.nist_800_53	AU.14, AC.7
t rule.pci_dss	10.2.4, 10.2.5
t rule.tsc	CC6.1, CC6.8, CC7.2, CC7.3
timestamp	Apr 27, 2022 @ 15:04:07.006

4. Wazuh rules and decoders

While Wazuh can receive and accept logs from many applications, it is not going to display all of them in the interface. If you have paid attention to the left side of the “discover” page, you may have found there are “categories” as shown in the following figure.



Actually only the events that are considered “alerts” by wazuh will be displayed. We can create “rules” to tell Wazuh what events should be considered as alerts and be displayed here.

To do this, in Wazuh manager, navigate to `/var/ossec/rulesets/rules`, you should see many pre-configured rules are listed in the folder. Those are the preset rules, once being matched, will trigger an alert in Wazuh.

Open the file `0540-pfsense_rules.xml`, and add the following code to the end of file but before the `</group>` tag.

```
<rule id="87703" level="5">
    <match>icmp</match>
    <description>pfsense ping detected.</description>

    <group>firewall_block,pci_dss_1.4,gpg13_4.12,hipaa_164.312.a.1,nist_800_53_SC.7,tsc_CC6.7,tsc_CC6.8,</group>
</rule>
```

This rule tells Wazuh if the log contains “icmp” keyword, it will trigger an alert with the given description.

Save the file, and then restart wazuh manager by `systemctl restart wazuh-manager`.

Provide a screenshot showing you have edited the file. – screenshot 3.



```
</rule>
<rule id="87703" level="5">
    <match>icmp</match>
    <description>pfsense ping detected.</description>
    <group>firewall_block,pci_dss_1.4,gpg13_4.12,hipaa_164.312.a.1,nist_800_53_SC.7,tsc_CC6.7,tsc_CC6.8,</group>
</rule>
</group>
```

5. Verify the rule works properly.

Login into pfsense, create a rule to block ICMP from LAN to WAN, remember to check “log packets that are handled by this rule” in the last section before you save the rule. This rule will block ping from LAN to WAN, and will log such events if it ever happens.

Do a ping from Admin to Server (5.5.5.5). If your rule is set correctly, the Ping should not get through. Wait 2 - 3 seconds, and then terminate the ping.

Login to Wazuh, navigate to Kibana → discover, and then search for keyword “icmp”, you should see several events shows that icmp were dropped.

Provide a screenshot shows you have successfully displayed this event in Wazuh. Zoom your web browser to make the font larger, and your screenshot should focus on the event and the content should be recognizable. – screenshot 4.



Table JSON

```
{
  "_id": "D-CDbIABVnNNnLgt0_F8",
  "_index": "wazuh-alerts-4.x-2022.04.27",
  "_score": -1,
  "_type": "_doc",
  "agent.id": "000",
  "agent.name": "wazuh-manager",
  "full_log": "Apr 27 19:33:38 filterlog[18300]: 86,,,1651087708,em1,match,block,in,4,0x0,,64,61168,0,DF,1,icmp,84,7.7.7.7,5.5.5.5,request,50217,764",
  "id": "1651088018.560370",
  "input.type": "log",
  "location": "7.7.7.1",
  "manager.name": "wazuh-manager",
  "predecoder.hostname": "filterlog[18300]:",
  "predecoder.timestamp": "Apr 27 19:33:38",
  "rule.description": "pfSense ping detected.",
  "rule.firedtimes": 43,
  "rule.gpg13": "4.12",
  "rule.groups": "pfsense, firewall_block, hipaa_164.312.a.1",
  "rule.id": "87703",
  "rule.level": 5,
  "rule.mail": false
}
```



```
t agent.name          wazuh-manager
t full_log            Apr 27 19:33:38 filterlog[18300]: 86,,,1651087708,em1,match,block,in,4,0x0,,64,61168,0,DF,1,icmp,84,
7.7.7.7,5.5.5.5,request,50217,764
t id                  1651088018.560370
t input.type          log
t location             7.7.7.1
t manager.name         wazuh-manager
t predecoder.hostname filterlog[18300]:
t predecoder.timestamp Apr 27 19:33:38
t rule.description     pfSense ping detected.
# rule.firedtimes      43
t rule.gpg13           4.12
t rule.groups          pfsense, firewall_block, hippa_164.312.a.1
t rule.id              87703
# rule.level           5
🔍 rule.mail            false
t rule.nist_800_53     SC.7
t rule.pci_dss         1.4
t rule.tsc             CC6.7, CC6.8
📅 timestamp           Apr 27, 2022 @ 15:33:38.342
```

This finishes this lab. Keep in mind that Wazuh is a fairly powerful SIEM that has much more potentials that you have learned in those labs. You can explore more functionalities from its user manual <https://documentation.wazuh.com/current/user-manual/overview.html> .