



wazuh.



Step-by-Step Guide: Automating Wazuh Security Alerts using n8n + Telegram

Created By: Parastou Razi

Wazuh → n8n → Telegram Automation

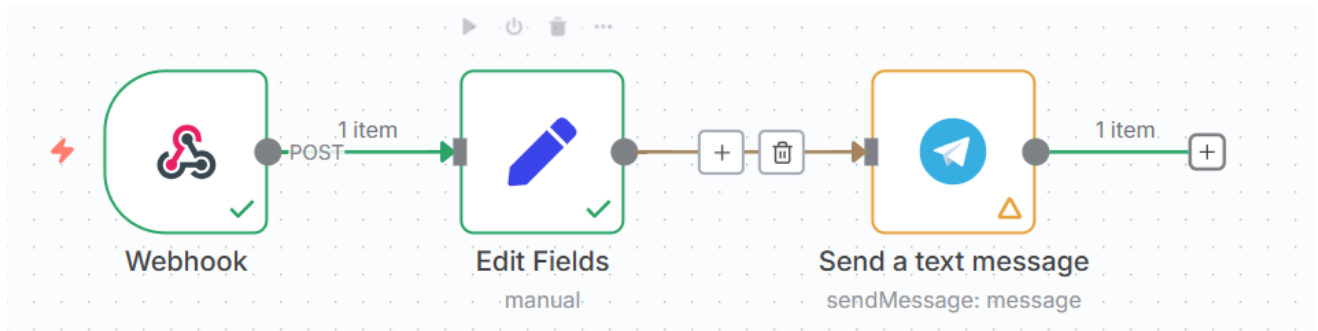
Tools Used

- **Wazuh** (SIEM - Security Information and Event Management)
- **n8n**
- **Telegram**

Objective

Automatically sending message to Telegram from Wazuh alerts

Workflow Overview



1. **Webhook** – Receives incoming Wazuh alert
2. **Edit Fields (Set)** – Extracts key data from the nested JSON body
3. **Send a Message (Telegram)**

Step-by-Step Setup

Wazuh Setup & Integration

1. Install Wazuh Manager (Ubuntu Example)

```
curl -sO https://packages.wazuh.com/4.12/wazuh-install.sh && sudo  
bash ./wazuh-install.sh -a
```

This installs the **Wazuh Manager**, **Dashboard**, and **Indexer** components.



2. Installing n8n

1. Download and install Node.js → <https://nodejs.org/en/download>

2. Open CMD and verify the installation:

`node -version`

3. Install n8n globally:

`npm install n8n -g`

4. Launch n8n and visit the dashboard:

`set N8N_SECURE_COOKIE=false`

`n8n`

Visit <http://localhost:5678>

3. OSSEC Integration Configuration

Add the following integration block to the `ossec.conf` file on the **Wazuh Manager** under the `<integration>` section:

`<integration>`

`<name>custom-n8n.py</name>`

`<hook_url>http://192.168.5.1:5678/webhook-test/e450257d-05e1-4881-8403-da4cb9425946</hook_url>`

`<rule_id>5501</rule_id>`

`<alert_format>json</alert_format>`

`</integration>`

4. Add Custom Python Script (Script is available in on my [GITHUB](#))


Place your script under:


`/var/ossec/integrations/`


Ensure it is executable:

`chmod +x /var/ossec/integrations/custom-n8n.py`

1. ☐ Webhook (Trigger)

 Webhook [Listen for test event](#)

[Parameters](#) [Settings](#) [Docs](#) 

 **Webhook URLs**


Test URL

Production URL

POST

http://localhost:5678/webhook-test/e450257d-05e1-4881-8403-da4cb9425946


HTTP Method

POST 


Path

e450257d-05e1-4881-8403-da4cb9425946

Authentication

None 

Respond


Immediately 


If you are sending back a response, add a "Content-Type" response header with the appropriate value to avoid unexpected behavior

Options


- **HTTP Method:** POST
- **Response Mode:** Immediately

2. Set (Edit Fields)

 **Edit Fields** Execute step


Parameters Settings Docs 


Mode

Manual Mapping 

Fields to Set

WazuhAlertID

T String 


= `{{ $json["body"]["id"] }}` 

1760636982.84282

Drag input fields here or **Add Field**


Include Other Input Fields
☒

Input Fields to Include

All 

Options


No properties


Add option 

Enable **“Include other input fields”**  Extract nested fields using expressions:



Label	Expression
Agent	<code>{{ \$json["body"]["all_fields"]["agent"]["name"] }}</code>
RuleID	<code>{{ \$json["body"]["rule_id"] }}</code>
Title	<code>{{ \$json["body"]["title"] }}</code>
Log	<code>{{ \$json["body"]["body"] }}</code>
Severity	<code>{{ \$json["body"]["severity"] }}</code>
Timestamp	<code>{{ \$json["body"]["timestamp"] }}</code>
WazuhAlertID	<code>{{ \$json["body"]["id"] }}</code>
Level (For IF)	<code>{{ \$json["body"]["all_fields"]["rule"]["level"] }}</code>

3. Sent To Telegram


 **Send a text message** Execute step

Parameters Settings Docs 


Credential to connect with

Telegram account  

Resource

Message 


Operation


Send Message 

Chat ID


138952878


Text




Subject:  Wazuh Alert - Tested By Parastou
astou

```
{{  
$json["body"]["all_fields"]["rule"]["description"]}}
```



Subject:  Wazuh Alert - Tested By Parastou PAM:...

Reply Markup

None 

Additional Fields

- **Subject:** 🚨 High Severity Wazuh Alert - Rule Description {{
- \$json["body"]["all_fields"]["rule"]["description"] }}
- **Body:**

🚨 ***High Severity Alert from Wazuh***

****Agent:**** {{ \$json["body"]["all_fields"]["agent"]["name"] }}

****Rule ID:**** {{ \$json["body"]["all_fields"]["rule"]["id"] }}

****Severity Level:**** {{ \$json["body"]["all_fields"]["rule"]["level"] }}

****Timestamp:**** {{ \$json["body"]["all_fields"]["timestamp"] }}

****Log:**** {{ \$json["body"]["all_fields"]["full_log"] }}

4. 📄 Testing Tips

- Use custom Python script to POST alert JSON
- Confirm webhook receives and parses alert

rule.description 🔍 »	rule.level	rule.groups	rule.id
sshd: authentication success.	3	syslog, sshd, authentication_success	5715
PAM: Login session opened.	3	pam, syslog, authentication_success	5501
sshd: authentication failed.	5	syslog, sshd, authentication_failed	5760
unix_chkpwd: Password check failed.	5	pam, syslog, authentication_failed	5557
PAM: User login failed.	5	pam, syslog, authentication_failed	5503
PAM: Login session closed.	3	pam, syslog	5502

🔧 Edit Fields Success in 2ms

OUTPUT

headers	params	query	body
host : 192.168.5.1:5678 user-agent : python-requests/2.25.1 accept-encoding : gzip, deflate accept : */* connection : keep-alive content-type : application/json accept-charset : UTF-8 content-length : 1422	{empty object}	{empty object}	severity : 1 low pretext : WAZUH Alert title : PAM: Login session opened. text : Oct 16 17:49:42 wazuh-server sshd[25499]: pam_unix(sshd:session): session opened for user root by (uid=0) rule_id : 5501 timestamp : 2025-10-16T17:49:42.503+0000 id : 1760636982.84282 body : Oct 16 17:49:42 wazuh-server sshd[25499]: pam_unix(sshd:session):

