GitHub Integration

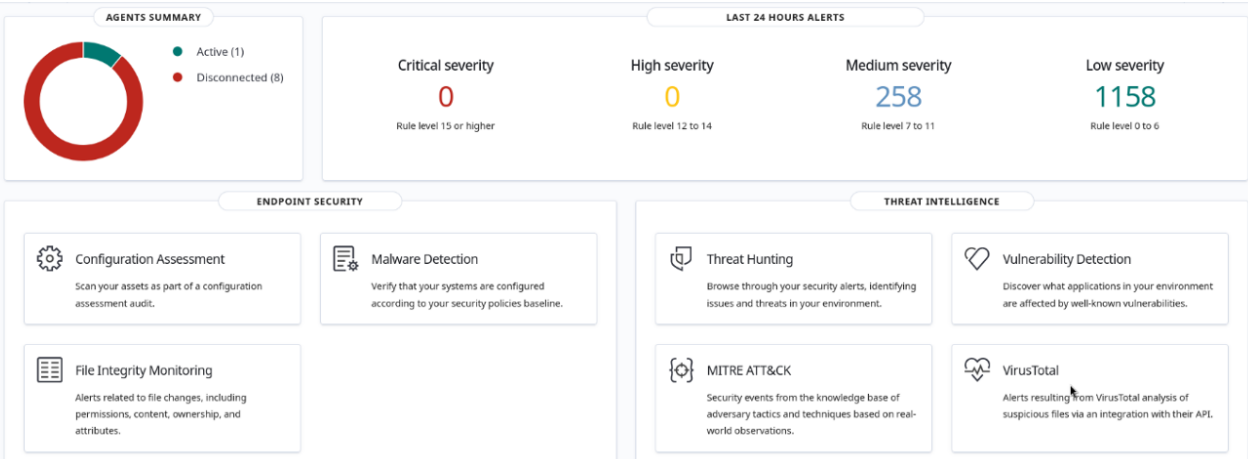
SAMEER HASSAN

Wazuh lab

GitHub-link: GitHub - sameerhassancode/Wazuh-labs

LinkedIn: https://www.linkedin.com/in/sameer-hassan-15a428255/

Wazuh-Dashboard:



AGENTS SUMMARY

● Active (1)
● Disconnected (8)

LAST 24 HOURS ALERTS

Critical severity	High severity	Medium severity	Low severity
0	0	258	1158
Rule level 15 or higher	Rule level 12 to 14	Rule level 7 to 11	Rule level 0 to 6

ENDPOINT SECURITY

Configuration Assessment
Scan your assets as part of a configuration assessment audit.

Malware Detection
Verify that your systems are configured according to your security policies baseline.

File Integrity Monitoring
Alerts related to file changes, including permissions, content, ownership, and attributes.

THREAT INTELLIGENCE

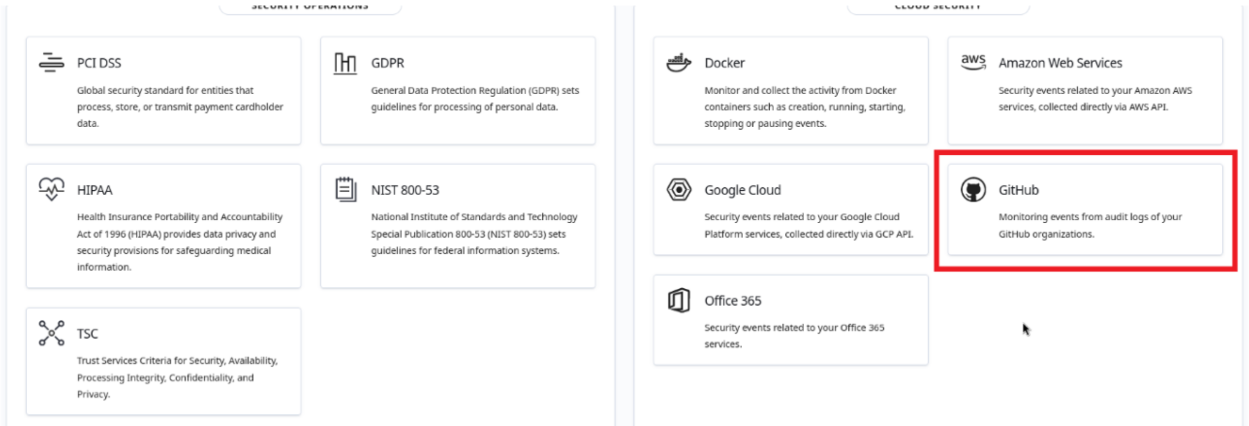
Threat Hunting
Browse through your security alerts, identifying issues and threats in your environment.

Vulnerability Detection
Discover what applications in your environment are affected by well-known vulnerabilities.

MITRE ATT&CK
Security events from the knowledge base of adversary tactics and techniques based on real-world observations.

VirusTotal
Alerts resulting from VirusTotal analysis of suspicious files via an integration with their API.

Now scroll down and look for GitHub



PCI DSS
Global security standard for entities that process, store, or transmit payment cardholder data.

GDPR
General Data Protection Regulation (GDPR) sets guidelines for processing of personal data.

HIPAA
Health Insurance Portability and Accountability Act of 1996 (HIPAA) provides data privacy and security provisions for safeguarding medical information.

NIST 800-53
National Institute of Standards and Technology Special Publication 800-53 (NIST 800-53) sets guidelines for federal information systems.

TSC
Trust Services Criteria for Security, Availability, Processing Integrity, Confidentiality, and Privacy.

Docker
Monitor and collect the activity from Docker containers such as creation, running, starting, stopping or pausing events.

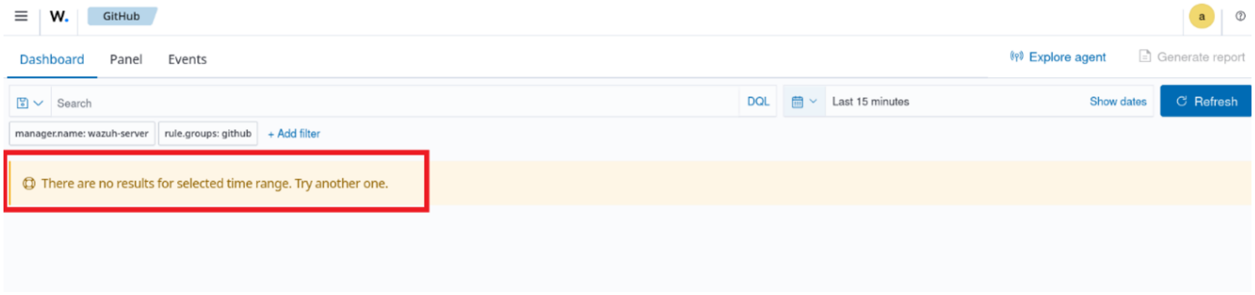
Amazon Web Services
Security events related to your Amazon AWS services, collected directly via AWS API.

Google Cloud
Security events related to your Google Cloud Platform services, collected directly via GCP API.

GitHub
Monitoring events from audit logs of your GitHub organizations.

Office 365
Security events related to your Office 365 services.

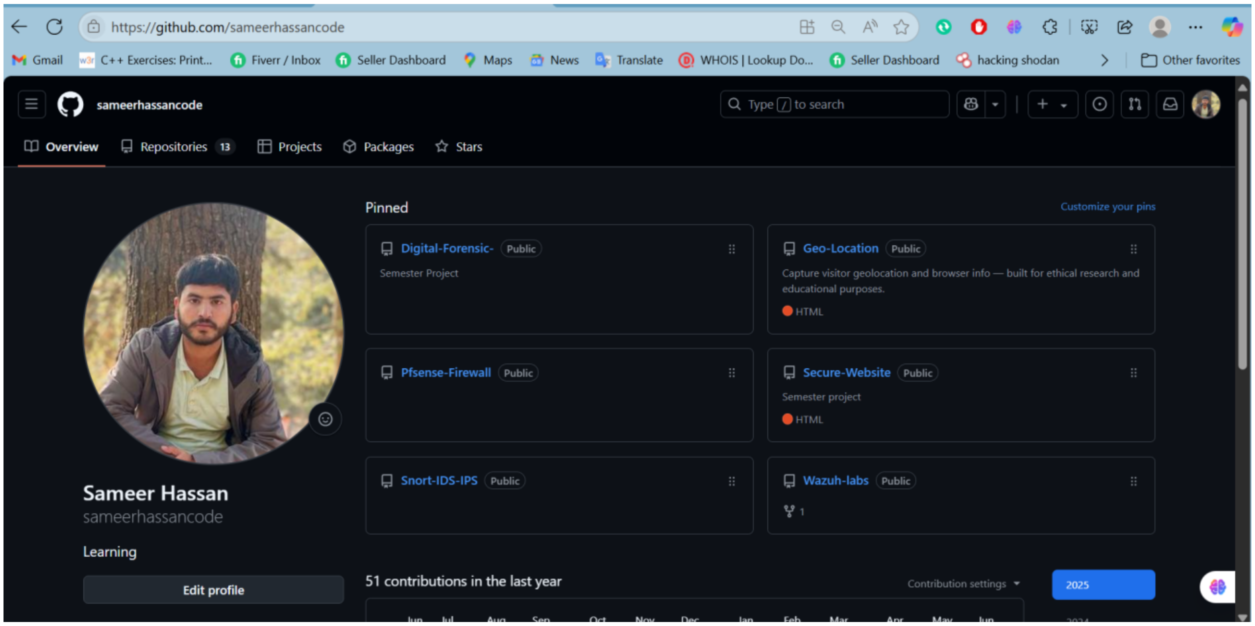
After opening the GitHub you will no logs



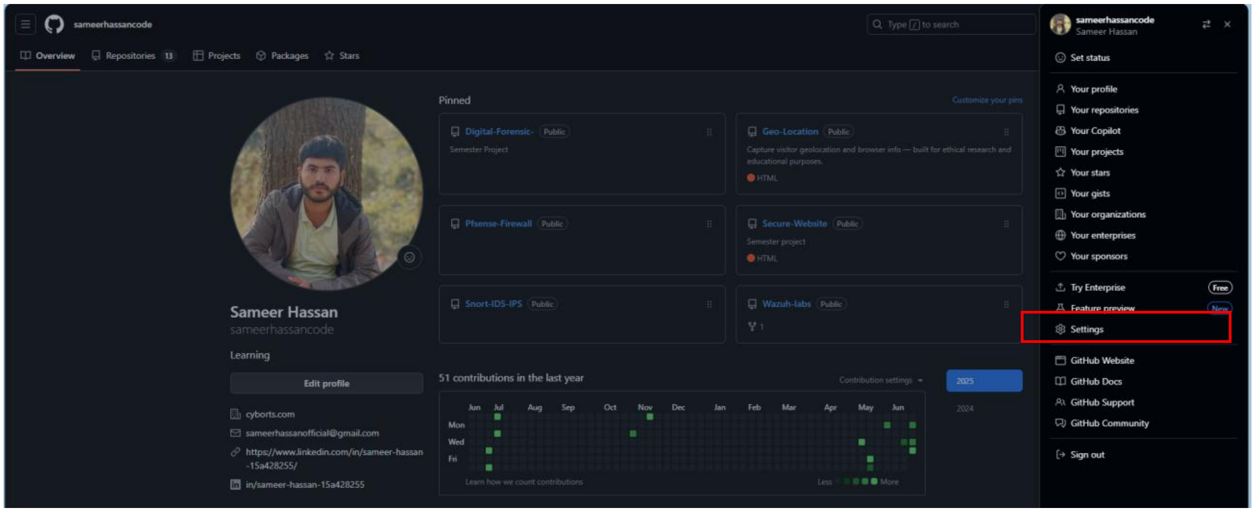
manager.name: wazuh-server | rule.groups: github

There are no results for selected time range. Try another one.

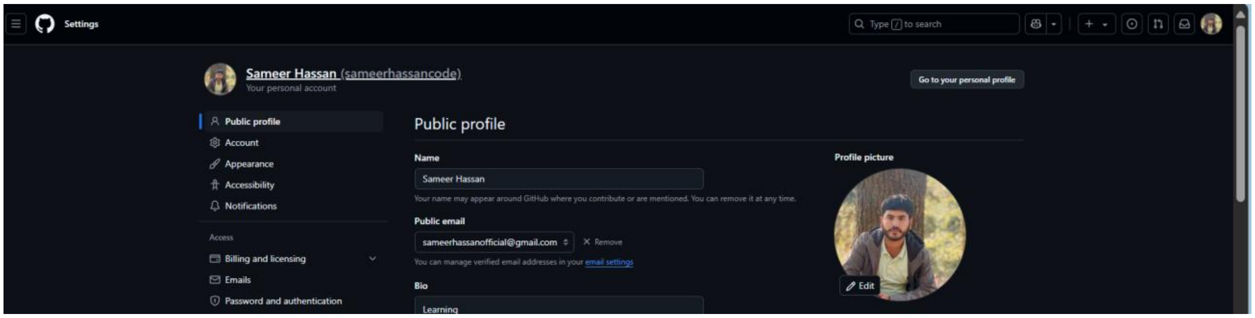
Now open your GitHub page in web browser



Now in the top right corner click on profile icon and you will see the setting



After clicking the setting button you will see this page. In security log you will see the log and all activities.



Public profile

Name

Sameer Hassan

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

Public email

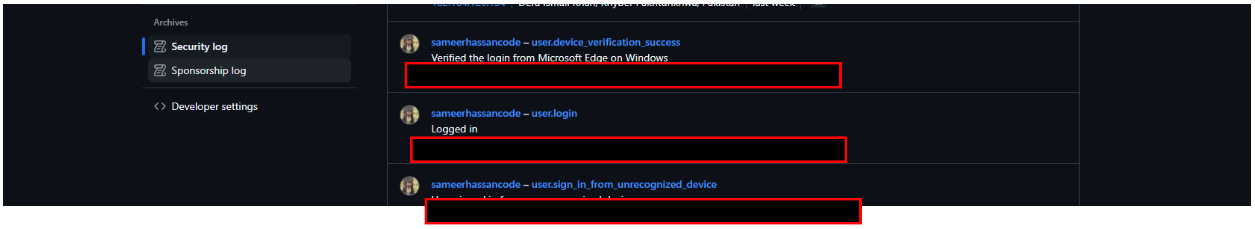
sameerhassanofficial@gmail.com ✕ Remove

You can manage verified email addresses in your email settings

Bio

Learning

Now scroll down and in left menu you will see the security logs

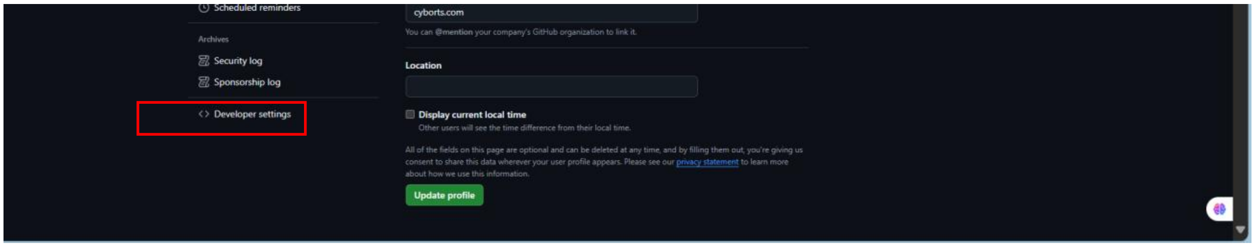


sameerhassancode – user.device_verification_success
Verified the login from Microsoft Edge on Windows
[redacted]

sameerhassancode – user.login
Logged in
[redacted]

sameerhassancode – user.sign_in_from_unrecognized_device
[redacted]

Now look for developer setting and click on it



cyborts.com

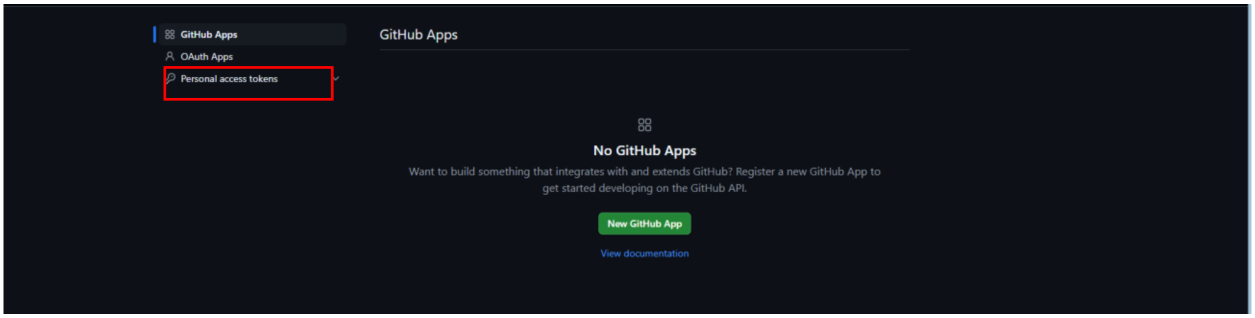
You can @mention your company's GitHub organization to link it.

Location

□ Display current local time
Other users will see the time difference from their local time.

All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our privacy statement to learn more about how we use this information.

After that you will able to see that page and click on **personal access token**

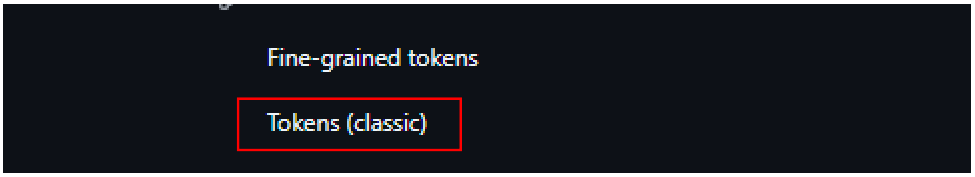


GitHub Apps

No GitHub Apps

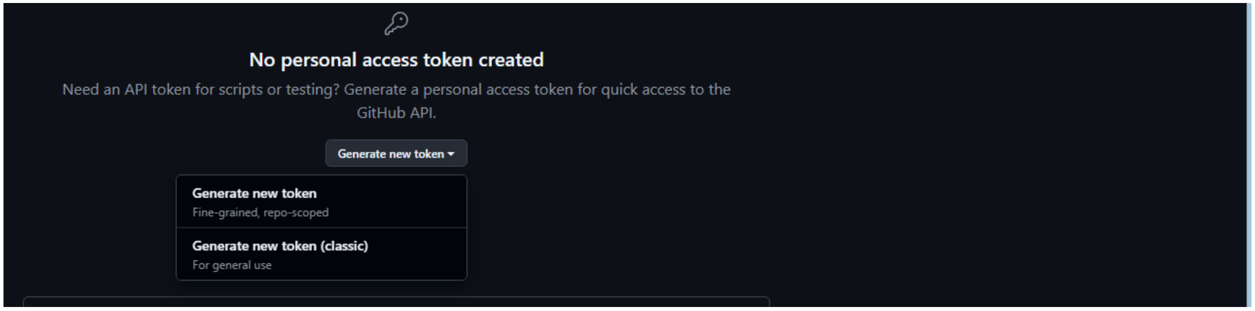
Want to build something that integrates with and extends GitHub? Register a new GitHub App to get started developing on the GitHub API.

Now again you have two options here chose Click token



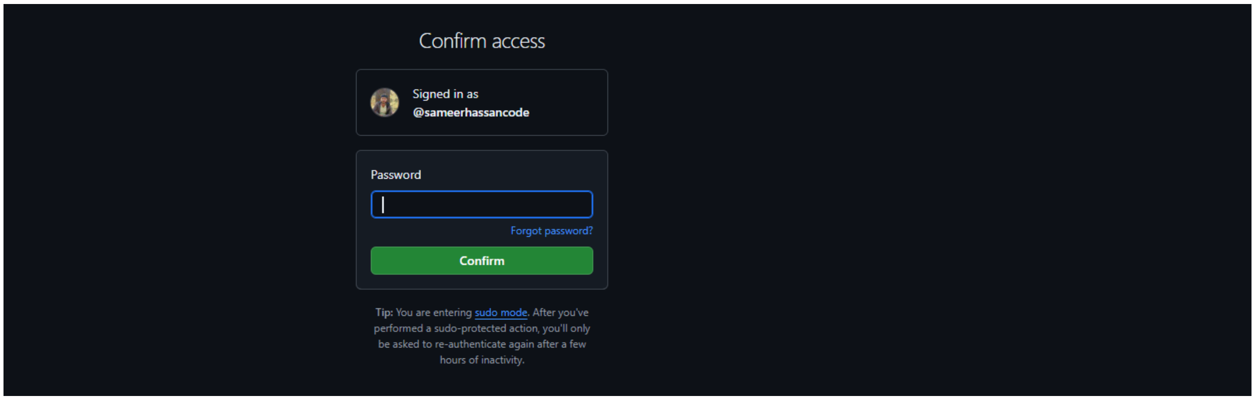
After clicking on **token (Classic)** you will see the button **Generate new token**

Then generate new token **(classic)**


No personal access token created

Need an API token for scripts or testing? Generate a personal access token for quick access to the GitHub API.

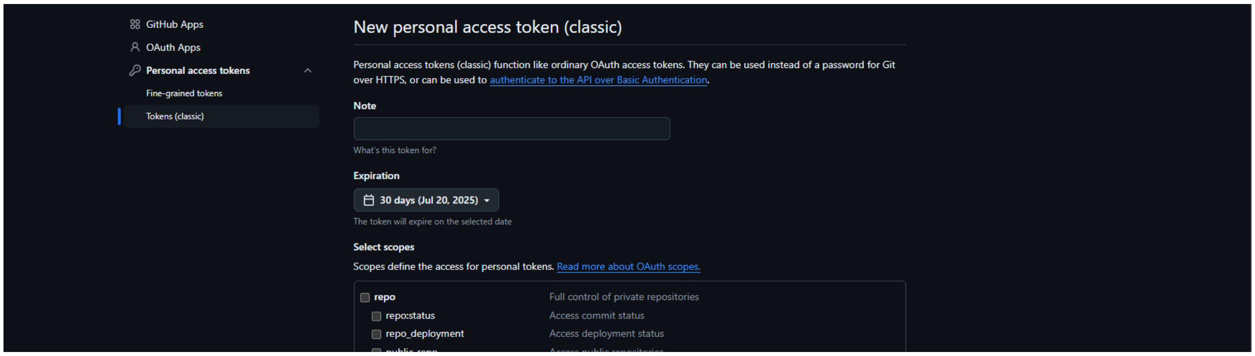
Now type you password here and click confirm


Confirm access

Signed in as **@sameerhassancode**

Tip: You are entering sudo mode. After you've performed a sudo-protected action, you'll only be asked to re-authenticate again after a few hours of inactivity.

After adding password you will redirect to this page here add detail like notes name expiration time and scope as per your need



New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

What's this token for?

Expiration

30 days (Jul 20, 2025) ▾

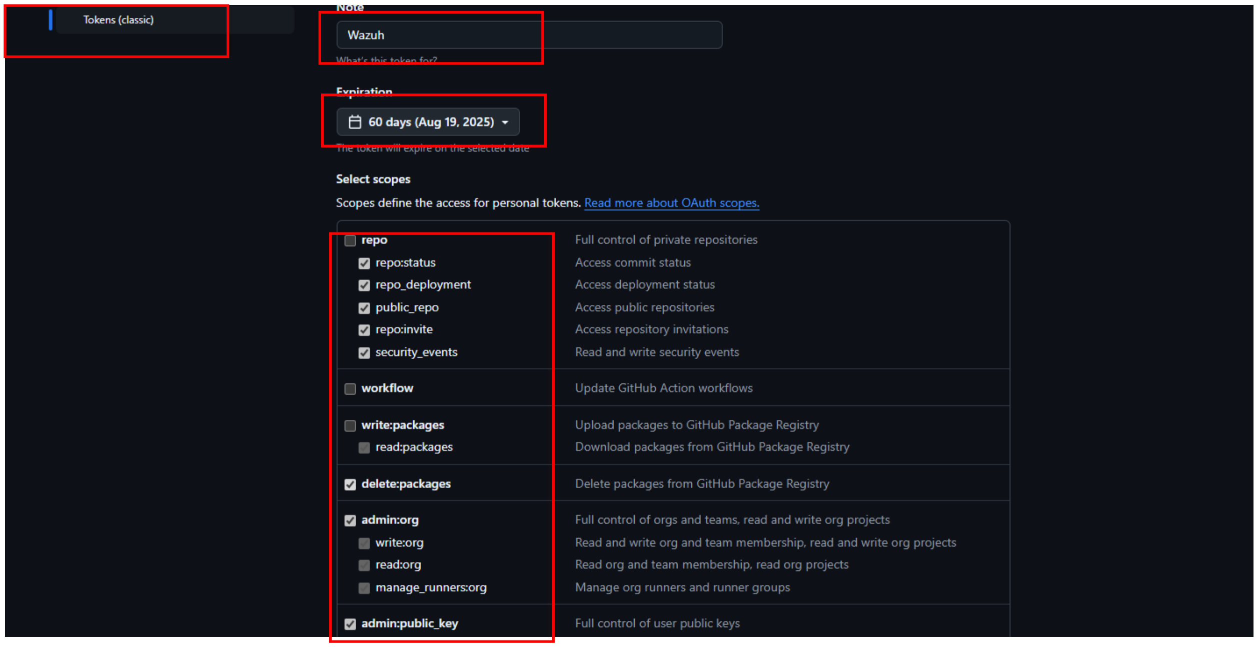
The token will expire on the selected date

Select scopes

Scopes define the access for personal tokens. Read more about OAuth scopes.

□ repo	Full control of private repositories
□ repo:status	Access commit status
□ repo_deployment	Access deployment status

Add and select detail.



Tokens (classic)

Note

Wazuh

What's this token for?

Expiration

60 days (Aug 19, 2025)

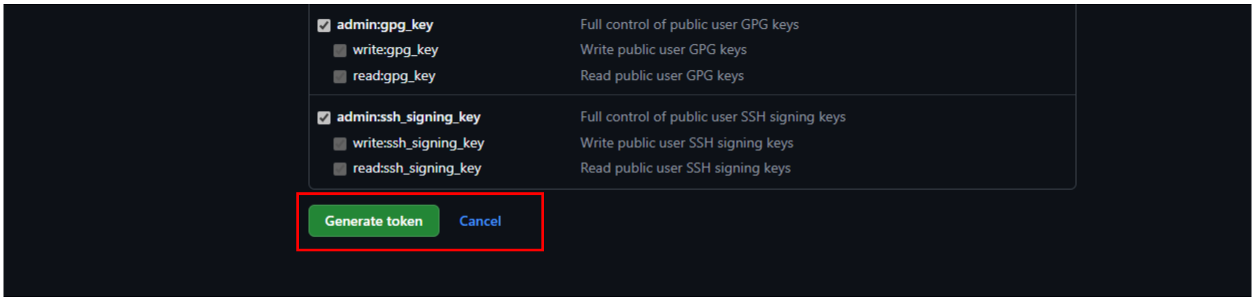
The token will expire on the selected date

Select scopes

Scopes define the access for personal tokens. Read more about OAuth scopes.

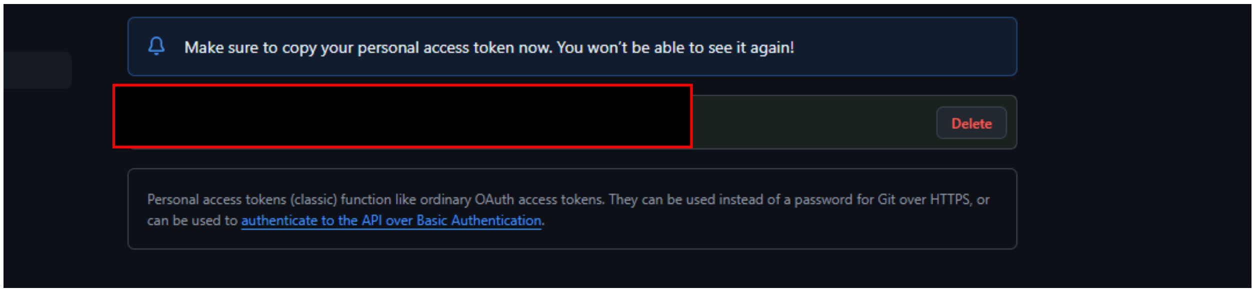
□ repo	Full control of private repositories
☑ repo:status	Access commit status
☑ repo_deployment	Access deployment status
☑ public_repo	Access public repositories
☑ repo:invite	Access repository invitations
☑ security_events	Read and write security events
□ workflow	Update GitHub Action workflows
□ write:packages	Upload packages to GitHub Package Registry
□ read:packages	Download packages from GitHub Package Registry
☑ delete:packages	Delete packages from GitHub Package Registry
☑ admin:org	Full control of orgs and teams, read and write org projects
□ write:org	Read and write org and team membership, read and write org projects
□ read:org	Read org and team membership, read org projects
□ manage_runners:org	Manage org runners and runner groups
☑ admin:public_key	Full control of user public keys

And click generate token



☑ admin:gpg_key	Full control of public user GPG keys
□ write:gpg_key	Write public user GPG keys
□ read:gpg_key	Read public user GPG keys
☑ admin:ssh_signing_key	Full control of public user SSH signing keys
□ write:ssh_signing_key	Write public user SSH signing keys
□ read:ssh_signing_key	Read public user SSH signing keys

It will redirect you to token page



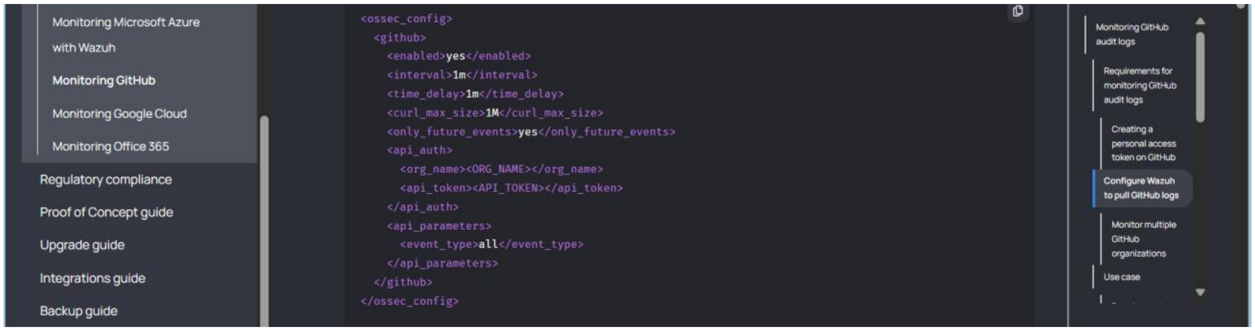
Make sure to copy your personal access token now. You won't be able to see it again!

[redacted]

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

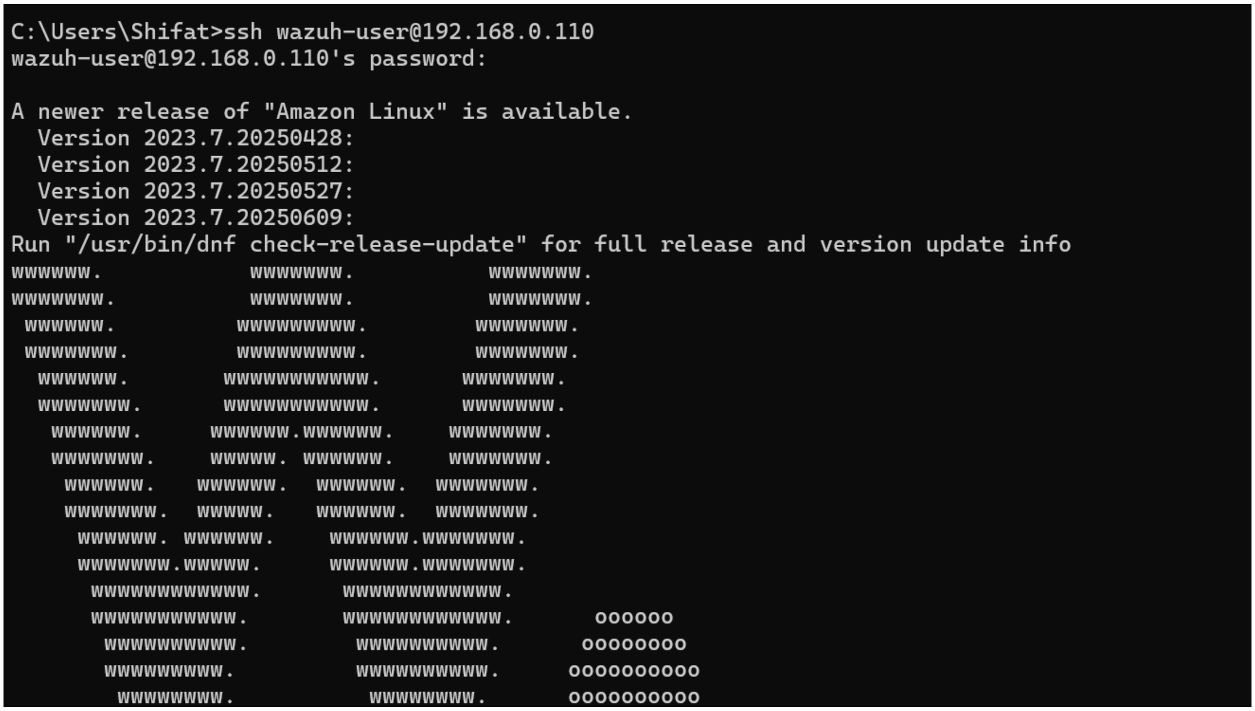
Now go to Wazuh documentation page and copy this code:

Link: Monitoring GitHub audit logs - Monitoring GitHub · Wazuh documentation


Monitoring Microsoft Azure with Wazuh
Monitoring GitHub
Monitoring Google Cloud
Monitoring Office 365
Regulatory compliance
Proof of Concept guide
Upgrade guide
Integrations guide
Backup guide
Monitoring GitHub audit logs
Requirements for monitoring GitHub audit logs
Creating a personal access token on GitHub
Configure Wazuh to pull GitHub logs
Monitor multiple GitHub organizations
Use case

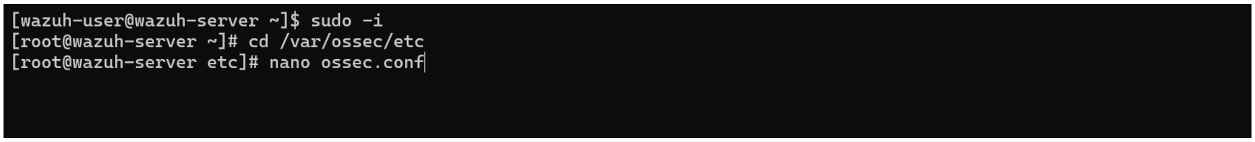
```
<ossec_config>
  <github>
    <enabled>yes</enabled>
    <interval>1m</interval>
    <time_delay>1m</time_delay>
    <curl_max_size>1M</curl_max_size>
    <only_future_events>yes</only_future_events>
    <api_auth>
      <org_name><ORG_NAME></org_name>
      <api_token><API_TOKEN></api_token>
    </api_auth>
    <api_parameters>
      <event_type>all</event_type>
    </api_parameters>
  </github>
</ossec_config>
```

After that start Wazuh server and connect with ssh



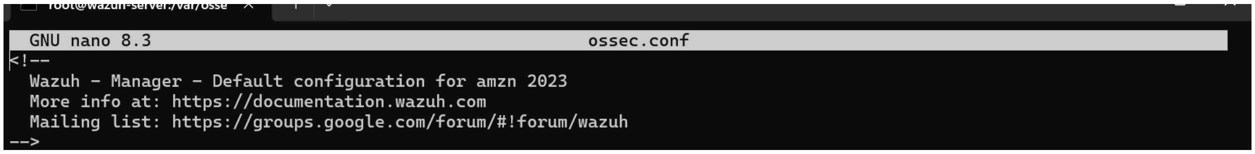
```
C:\Users\Shifat>ssh wazuh-user@192.168.0.110
wazuh-user@192.168.0.110's password:

A newer release of "Amazon Linux" is available.
  Version 2023.7.20250428:
  Version 2023.7.20250512:
  Version 2023.7.20250527:
  Version 2023.7.20250609:
Run "/usr/bin/dnf check-release-update" for full release and version update info
wwwwww.           wwwwwww.          wwwwwww.
wwwwwww.          wwwwwww.          wwwwwww.
 wwwwww.         wwwwwwwww.        wwwwwww.
 wwwwwww.        wwwwwwwww.        wwwwwww.
  wwwwww.       wwwwwwwwwww.      wwwwwww.
  wwwwwww.      wwwwwwwwwww.      wwwwwww.
   wwwwww.     wwwwww.wwwwww.    wwwwwww.
   wwwwwww.    wwwww. wwwwww.    wwwwwww.
    wwwwww.   wwwwww.  wwwwww.  wwwwwww.
    wwwwwww.  wwwww.   wwwwww.  wwwwwww.
     wwwwww. wwwwww.    wwwwww.wwwwwww.
     wwwwwww.wwwww.     wwwwww.wwwwwww.
      wwwwwwwwwwww.      wwwwwwwwwwww.
      wwwwwwwwwww.       wwwwwwwwwwww.      oooooo
       wwwwwwwwww.        wwwwwwwwww.      oooooooo
       wwwwwwwww.         wwwwwwwwww.     oooooooooo
        wwwwwwww.          wwwwwwww.      oooooooooo
```

Commands: **sudo -i** then **cd /var/ossec/etc**



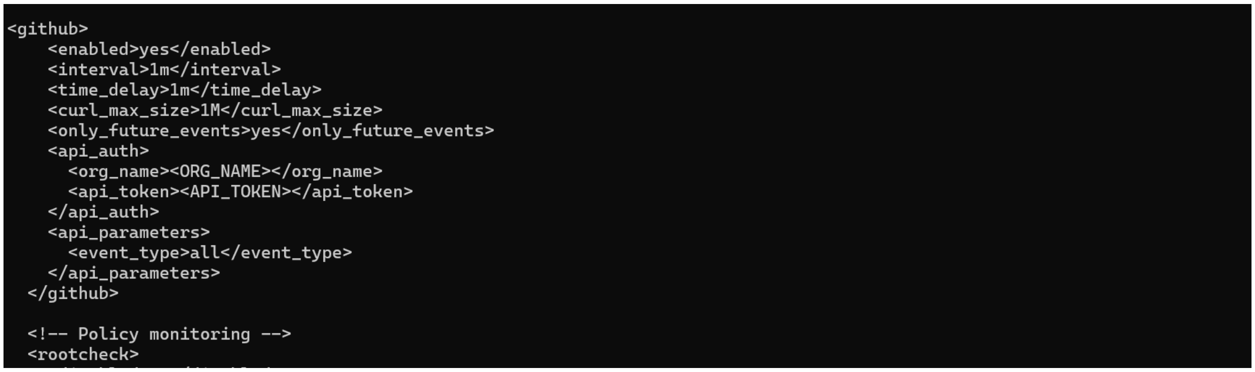
```
[wazuh-user@wazuh-server ~]$ sudo -i
[root@wazuh-server ~]# cd /var/ossec/etc
[root@wazuh-server etc]# nano ossec.conf
```

open file with **nano ossec.conf** command



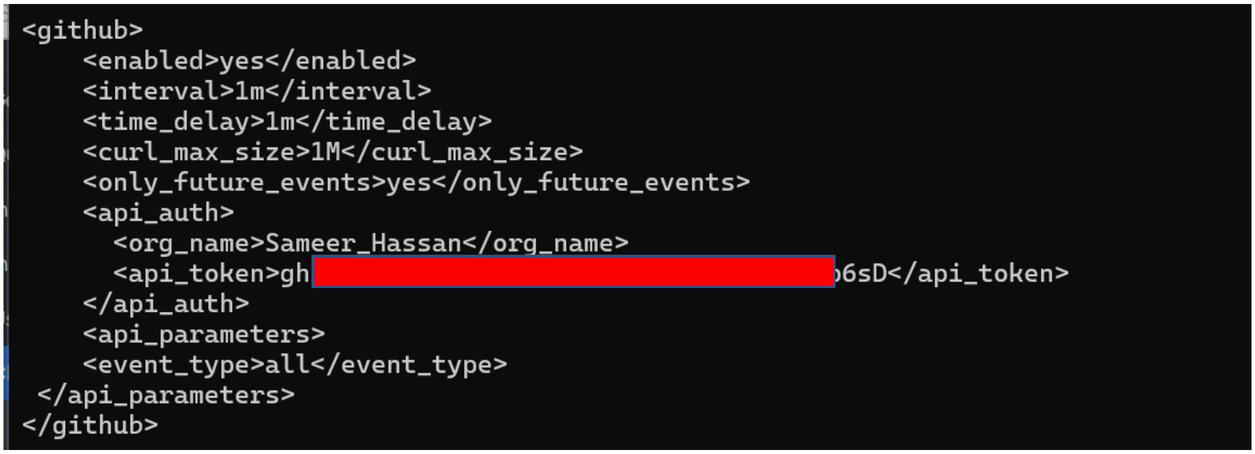
```
  GNU nano 8.3                                    ossec.conf
<!--
  Wazuh - Manager - Default configuration for amzn 2023
  More info at: https://documentation.wazuh.com
  Mailing list: https://groups.google.com/forum/#!forum/wazuh
-->
```

And then add this code



```
<github>
    <enabled>yes</enabled>
    <interval>1m</interval>
    <time_delay>1m</time_delay>
    <curl_max_size>1M</curl_max_size>
    <only_future_events>yes</only_future_events>
    <api_auth>
      <org_name><ORG_NAME></org_name>
      <api_token><API_TOKEN></api_token>
    </api_auth>
    <api_parameters>
      <event_type>all</event_type>
    </api_parameters>
  </github>

  <!-- Policy monitoring -->
  <rootcheck>
```

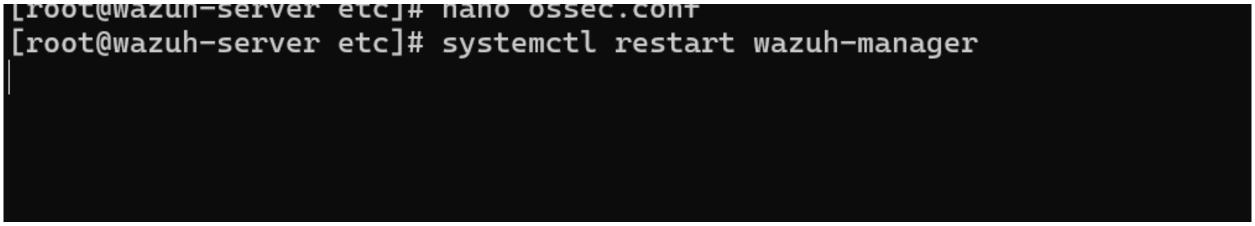
Replace the ORG_name with any name you want and add api_token



```
<github>
    <enabled>yes</enabled>
    <interval>1m</interval>
    <time_delay>1m</time_delay>
    <curl_max_size>1M</curl_max_size>
    <only_future_events>yes</only_future_events>
    <api_auth>
      <org_name>Sameer_Hassan</org_name>
      <api_token>gh[redacted]6sD</api_token>
    </api_auth>
    <api_parameters>
    <event_type>all</event_type>
 </api_parameters>
</github>
```

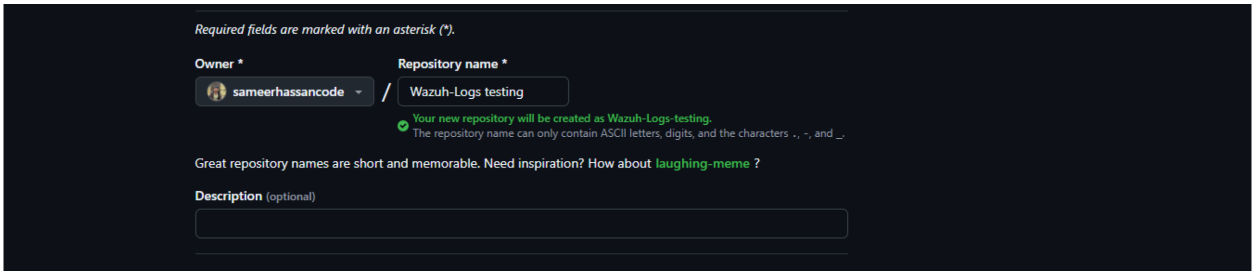
Save the file and restart the Wazuh-manager

Systemclt restart Wazuh-manager



```
[root@wazuh-server etc]# nano ossec.conf
[root@wazuh-server etc]# systemctl restart wazuh-manager
```

Now again open a Github and create or delete some file for logs



Required fields are marked with an asterisk (*).

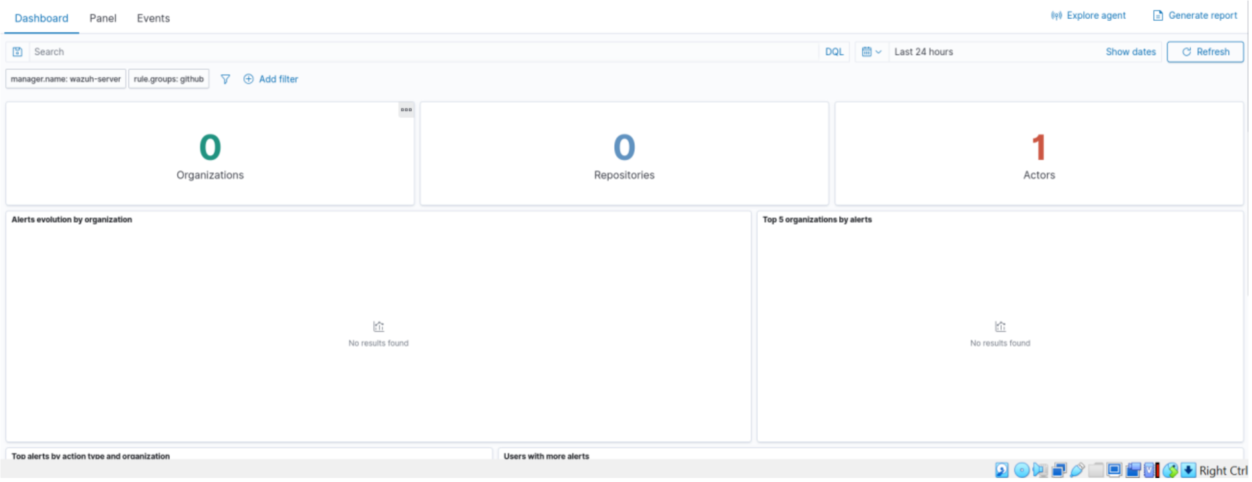
Owner * sameerhassancode / Repository name * Wazuh-Logs testing

Your new repository will be created as Wazuh-Logs-testing.
The repository name can only contain ASCII letters, digits, and the characters ., -, and _.

Great repository names are short and memorable. Need inspiration? How about laughing-meme ?

Description (optional)

Open Wazuh dashboard and see the logs of newly created and deleted directory



manager.name: wazuh-server | rule.groups: github

0 Organizations

0 Repositories

1 Actors

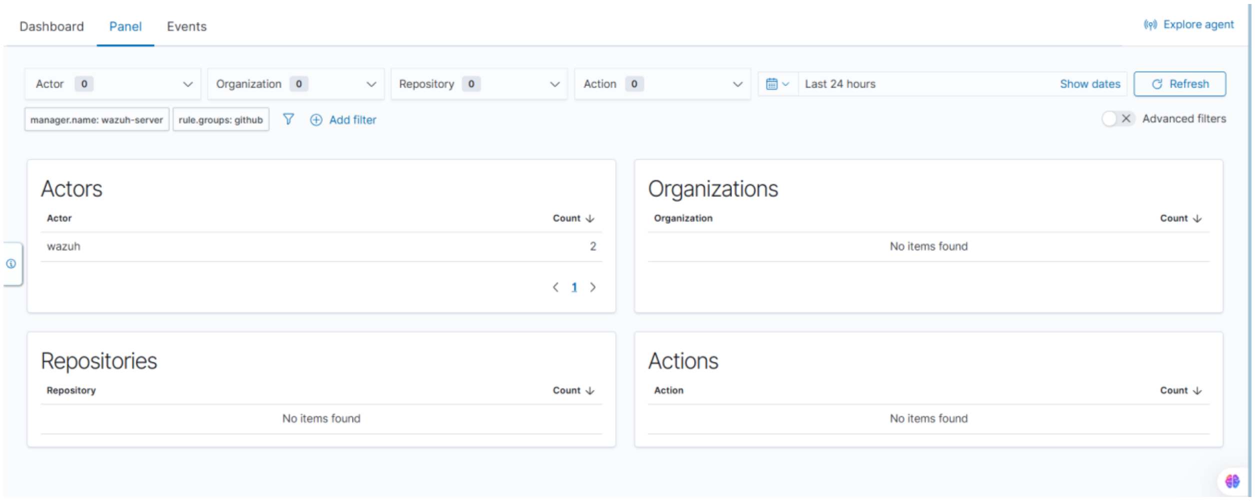
Alerts evolution by organization: No results found

Top 5 organizations by alerts: No results found

Top alerts by action type and organization

Users with more alerts

Panel section:



Actor 0 | Organization 0 | Repository 0 | Action 0

manager.name: wazuh-server | rule.groups: github

Actors

Actor	Count
wazuh	2

Organizations

Organization	Count
No items found	

Repositories

Repository	Count
No items found	

Actions

Action	Count
No items found	

Perform some action and Wazuh will get the logs from the GitHub

Summary:

Integrating Wazuh with GitHub significantly enhances an organization's DevSecOps strategy by securing code repositories and ensuring compliance with internal standards.

Need training on Wazuh ?

Contact number: +923355345678

Email: sameeerishassan@gmail.com

LinkedIn: https://pk.linkedin.com/in/sameer-hassan-15a428255

Other SIEM

1. IBM Qradar

2. Splunk

3. Azure Sentinel