Threat Fusion:

Wazuh + TheHive SOAR Integration

PROJECT REPORTED BY:

VISHWA PANCHOLI

PROJECT GUIDED BY:

POOJA MATHUR

ENROLMENT NO: 202118100092

DEPARTMENT OF ANIMATION ,

ITIMS & MOBILE APPLICATIONS

GUJARAT UNIVERSITY

BATCH 2021-2024

GUJARAT UNIVERSITY

Department of Animation, ITIMS & Mobile Applications

Certificate

Enrolment No: 202118100092

This is to certify that Ms. Vishwa Pancholi student of MSc. IT IMS & CS (Integrated) Semester – 6, has duly completed her Term work for the semester ending in May 2024, in the subject of Project towards partial fulfilment of her Degree of Bachelor Program.

Date of Submission :
18/05/2024

Mentor
Mrs. Pooja Mathur

Index

Acknowledgement

I would like to express my sincere gratitude to all those who have contributed to the realization of this project. Firstly, I extend my deepest appreciation to my mentor and to my department their expertise and encouragement have been invaluable in shaping the direction and scope of this endeavour.

I am also indebted to my colleagues and peers for their collaboration and assistance. Their insights, feedback, and constructive criticism have significantly enriched the project, driving it towards greater refinement and effectiveness.

Furthermore, I would like to acknowledge the contributions of the open-source community, whose innovative tools and resources have formed the foundation of this project. Their commitment to knowledge sharing and collaboration has been instrumental in advancing the field of cybersecurity.

Last but not least, I am grateful to my family and friends for their patience, understanding, and encouragement during the course of this project. Their unwavering support has been a constant source of motivation and inspiration.

Student Name: Vishwa Pancholi

Enrollment Number: 202118100092

Preface

Welcome to the documentation of "ThreatFusion: Wazuh + TheHive SOAR Integration." this project represents an innovative approach to enhancing cybersecurity operations through the integration of Wazuh and TheHive SOAR platforms. By combining these powerful tools, the aim is to create a cohesive workflow that facilitates the detection and response to security threats in a timely and efficient manner.

The primary objective of this documentation is to provide a comprehensive overview of the project, including its objectives, methodologies, and implementation strategies. Through detailed explanations and practical examples, readers will gain insight into the intricacies of the integrated solution and learn how to leverage its capabilities to enhance their own security operations.

Whether you are a seasoned cybersecurity professional or a newcomer to the field, this documentation is designed to be accessible and informative. It serves as a valuable resource for anyone seeking to understand the intricacies of threat detection and response, as well as the integration of soar platforms into existing security infrastructure.

I hope that this documentation will serve as a useful guide and inspire further exploration and innovation in the field of cybersecurity.

1. Introduction

- Cybersecurity is increasingly important as threats become more sophisticated. "ThreatFusion: Wazuh + TheHive SOAR Integration" is a project that combines two powerful cybersecurity tools, Wazuh and TheHive with the Shuffler to make organizations safer online.

- In this intro, we'll explain what the project aims to achieve and why it's important. We'll also briefly touch on what Wazuh and TheHive are and how they work together to keep organizations secure.

- Let's get started on this journey to understand how this integration strengthens cybersecurity in simple terms.

1.1 Introduction of Wazuh

- Wazuh is a comprehensive open-source cybersecurity platform that provides threat detection, integrity monitoring, and security analytics capabilities. It is designed to help organizations enhance their security posture and protect against a wide range of cyber threats.

- Wazuh operates by collecting and analysing data from various sources within the IT environment, including logs, events, and configurations. This data is processed in real-time using a combination of predefined rules, machine learning algorithms, and anomaly detection techniques. When suspicious activity is detected, Wazuh generates alerts, initiates response actions, and provides detailed information for investigation by security analysts.

Architecture Model & Components :

- Wazuh follows a distributed architecture model, consisting of three main components:

i. WAZUH MANAGER :
   - The central component of the Wazuh architecture, the Wazuh Manager is responsible for data collection, analysis, and management. It serves as the centralized server that receives data from Wazuh Agents, processes it, and generates alerts.

ii. WAZUH AGENT :
   - Wazuh Agents are lightweight software agents installed on monitored systems, including servers, workstations, and cloud instances. These agents collect security-relevant data from the host system, such as logs, events, file integrity changes, and system configurations. They then forward this data to the Wazuh Manager for analysis.

iii. ELASTIC STACK :
   - Wazuh integrates with the Elastic Stack, which includes Elasticsearch, Logstash, and Kibana (ELK stack), for data storage, processing, and visualization. Elasticsearch is used for indexing and searching security data, Logstash for data parsing and enrichment, and Kibana for data visualization and analysis.

Features :

- Log and File Integrity Monitoring :
  Monitors changes to logs and critical files for signs of tampering, unauthorized access, or suspicious activities.

- Intrusion Detection :
  Detects and alerts on known attack patterns, anomalies, and suspicious behaviour indicative of security threats, such as brute-force attacks, malware infections, and command and control communication.

- Vulnerability Detection :
  Identifies security weaknesses, misconfigurations, and vulnerabilities in the IT infrastructure that could be exploited by attackers.

- Compliance Monitoring :
  Helps ensure adherence to security standards, regulations, and internal security policies by monitoring for policy violations, configuration drift, and non-compliance.

- Incident Response :
  Facilitates rapid response to security incidents by providing actionable alerts, response capabilities, and integration with incident response workflows.

Services :

- Threat Detection :
  Provides real-time detection and alerting on security threats, allowing for timely response and mitigation to minimize the impact of security incidents.

- Compliance Monitoring :
  Helps organizations maintain compliance with industry regulations, standards, and internal security policies by monitoring for violations and non-compliance.

- Incident Response :
  Enables efficient incident response by providing detailed information, context, and response actions to security incidents, facilitating rapid containment and remediation.

- Log Analysis :
  Analyses logs and events from various sources, including hosts, network devices, and applications, to identify patterns, trends, and anomalies indicative of security threats, vulnerabilities, or compliance issues.

- Wazuh's comprehensive set of features and services make it a valuable tool for organizations looking to enhance their cybersecurity posture, improve threat detection capabilities, and protect against evolving cyber threats.

1.2 Introduction of TheHive

- TheHive is an open-source Security Orchestration, Automation, and Response (SOAR) platform designed to facilitate incident response and threat intelligence activities within organizations. It enables security teams to effectively manage and respond to security incidents, streamline workflows, and automate response actions.

- TheHive works by aggregating and correlating security-related data from various sources, including SIEM systems, threat intelligence feeds, and incident reports. It provides a centralized platform for security analysts to triage, investigate, and respond to security incidents efficiently. TheHive's flexible architecture allows for integration with external tools and services, enabling seamless collaboration and automation of response actions.

Architecture Model & Components :

- TheHive Server :
  The central component of TheHive architecture, TheHive Server is responsible for managing incidents, users, and case-related data. It provides a web-based interface for security analysts to interact with and perform various incident response tasks.

- Case Management :
  TheHive's case management functionality allows security teams to organize and track security incidents throughout their lifecycle, from initial triage to final resolution. Cases can be assigned to individual analysts, annotated with relevant information, and linked to related incidents or observables.

- Alert Aggregation :
  TheHive aggregates alerts and notifications from various sources, such as SIEM systems, threat intelligence feeds, and automated detection tools. It correlates these alerts to create unified incident records, providing a holistic view of security incidents and their associated context.

Features :

- Incident Triage :
  Enables security teams to quickly triage and prioritize security incidents based on severity, impact, and urgency.

- Investigation Workflow :
  Provides customizable workflows for conducting thorough investigations, including tasks, artifacts, and collaboration features.

- Response Automation :
  Automates response actions and playbooks to facilitate rapid containment, mitigation, and remediation of security incidents.

- Evidence Management :
  Allows for the collection, preservation, and analysis of digital evidence related to security incidents, ensuring chain of custody and integrity.

- Integration Framework :
  Offers a rich set of integrations with external tools and services, including SIEM systems, threat intelligence platforms, and forensic analysis tools.

Services :

- Incident Management :
  Facilitates centralized incident management, tracking, and reporting for security teams, providing visibility into incident status and progress.

- Collaboration :
  Enables seamless collaboration and communication among security analysts, allowing for real-time information sharing and decision-making.

- Automation :
  Automates repetitive tasks and response actions to improve efficiency and reduce response times, freeing up analysts to focus on higher-value activities.

- Intelligence Sharing :
  Facilitates the sharing of threat intelligence and best practices among security teams, helping to enhance overall situational awareness and response capabilities.

- TheHive's comprehensive set of features and services make it a powerful tool for security operations teams looking to improve incident response capabilities, streamline workflows, and mitigate security risks effectively.

1.3 Project Overview

- The goal of the project "Threatfusion: Wazuh + TheHive SOAR Integration" is to create an efficient cybersecurity solution. This solution revolves around defining a practical workflow where a specific tool called Mimikatz, when used on Windows agents, is detected by the system. Once detected, an automated alert is generated and sent to a Security Operations Center (SOC) Analyst via email.

- The integration of Wazuh and TheHive SOAR holds significant importance for enhancing cybersecurity capabilities. By automating the detection and response to security threats, organizations can improve their ability to protect sensitive information, mitigate risks, and maintain operational continuity.

- Integration of Wazuh and TheHive : The project focuses on seamlessly integrating Wazuh, a security monitoring platform, with TheHive, a Security Orchestration, Automation, and Response (SOAR) platform. This integration enables the automation of incident response processes.

- Workflow Development : A key aspect of the project is the development of a workflow that detects the usage of Mimikatz on Windows agents. This workflow is designed to trigger automated actions, including alert generation and email notification to SOC Analysts, upon detection of suspicious activity.

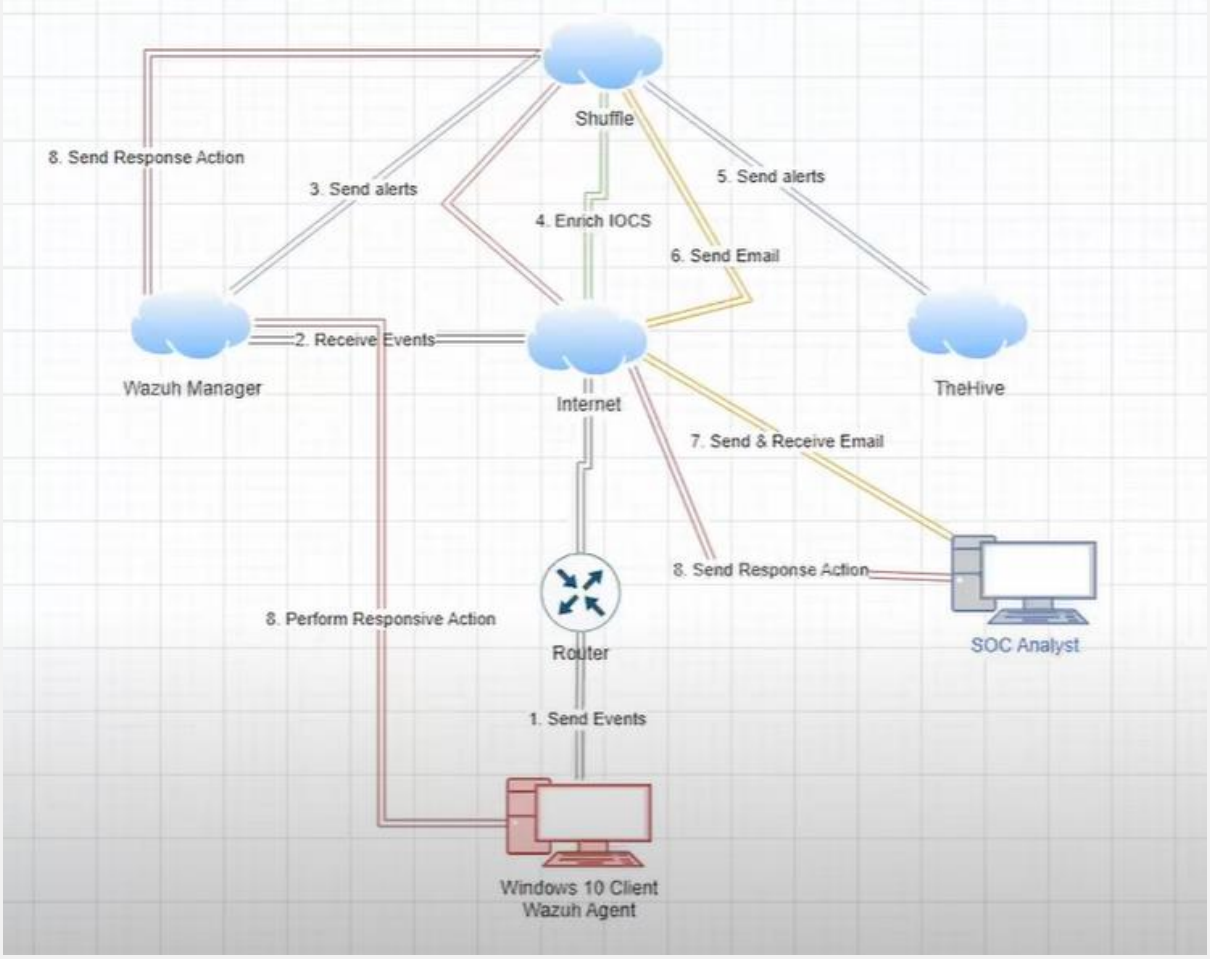
- The project scope includes:
    - Configuring and deploying Wazuh and TheHive SOAR platforms.
    - Developing a custom workflow to detect the usage of Mimikatz on Windows agents.
    - Implementing automated actions, such as alert generation and email notification, upon detection of suspicious activity.
    - Testing and validating the integrated solution to ensure its effectiveness and reliability.

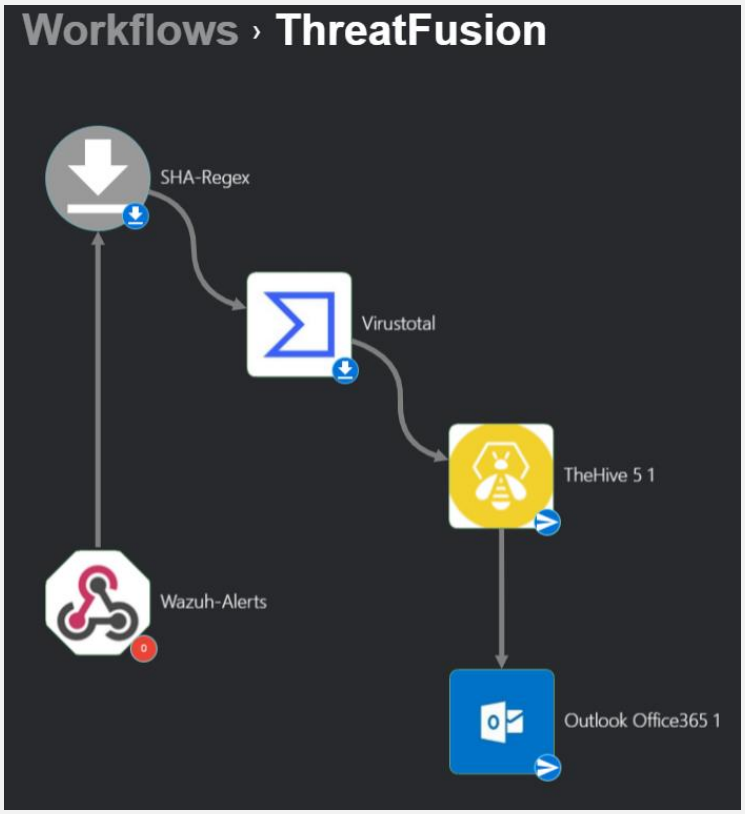
- This project aims to empower organizations with a proactive approach to cybersecurity, enabling them to detect and respond to security threats in a timely and efficient manner.

2. Lab Design & System Requirements

Lab Design :

- The lab design illustrates the network architecture, server setup, and integration configuration for the Threatfusion project. It provides a visual representation of how the various components interact within the lab environment.





System Requirements :

- Here's the Hardware and Software Requirements for setting up the lab environment, including server specifications, Operating Systems.

- Hardware Requirements :

  - Laptop or PC

  - RAM = 8Gb Minimum ; 16 Gb for better performance

  - Enough Storage Space

Software Requirements :

- Google Cloud Platform Free Account ; If you are developing in Cloud otherwise you can setup in you VM as well.
- VMware Workstation
- Operating System : Ubuntu or RedHat for Wazuh & TheHive and Windows 10,11 for Agent

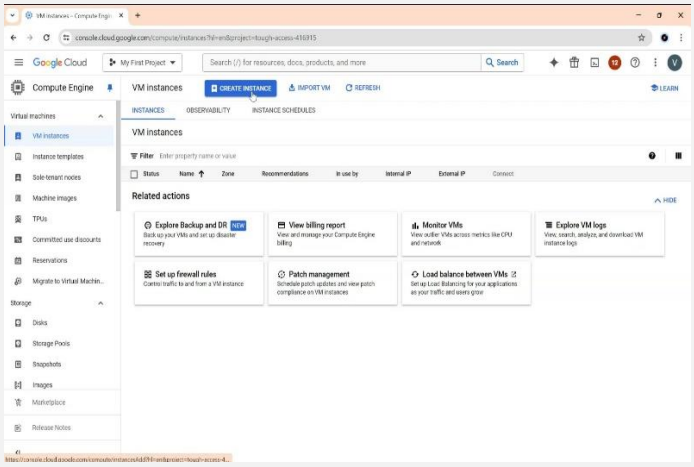
3. Setting Up the Wazuh

- Installing Wazuh and adding agents is a crucial step in setting up the lab environment for the Threatfusion project. This section outlines the process for installing Wazuh on the central server and then adding agents to monitor target Windows systems.

3.1 Installation of Wazuh

- For the installation of Wazuh, you have the flexibility to deploy the Wazuh Manager on various platforms, including virtual machines running Ubuntu on VMware or cloud platforms like GCP, Azure, AWS. Here, we'll guide you through the process of installing Wazuh on an Ubuntu virtual machine in Google Cloud Platform(GCP).

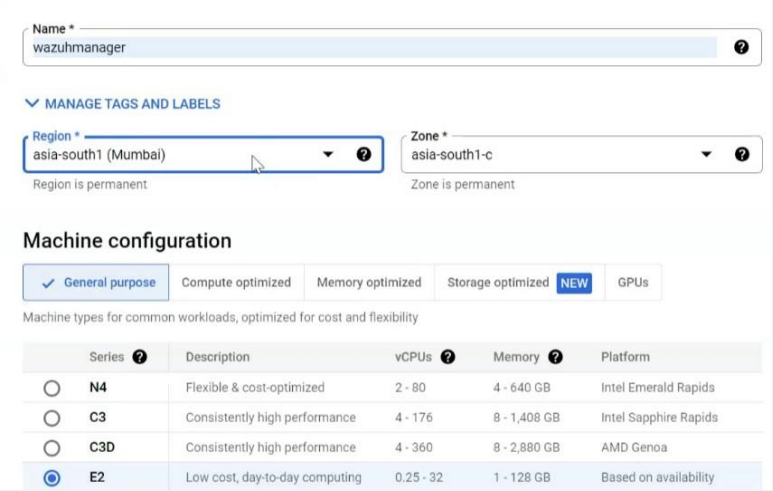
- Next, I'll install Wazuh on Google Cloud Platform (GCP). I have a free account on GCP, and you'll need one too if you want to follow along. If you need to create an account, make sure you have access to a free account on GCP as well.



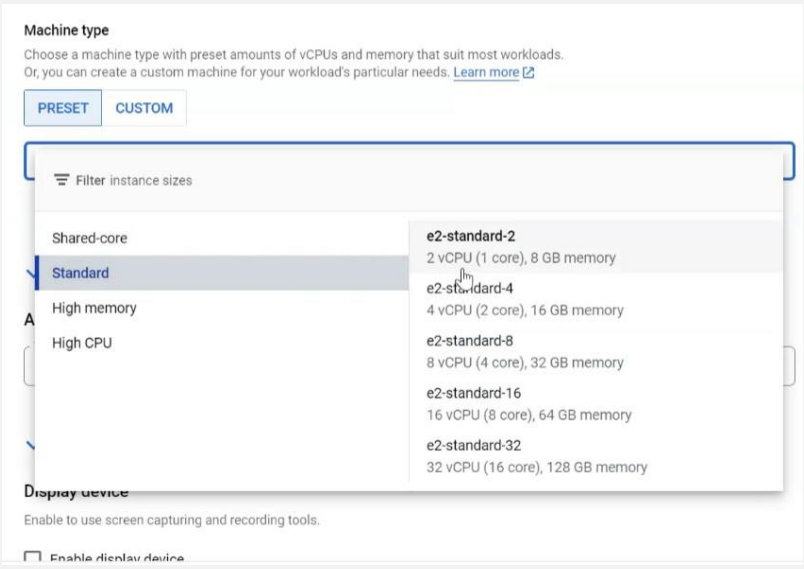
This is our GCP Console View. Here I'm in Compute Engine -> VM Instances.

- Now we are going to follow below steps to Create the VM for our Wazuh Manager.

- Step 1:
  - We are going to Make VM for that Click on Create Instance.
  - Now, Here We are giving name to our VM as a *wazuhmanager*. and also, we are selecting our Region *Asia-south*.

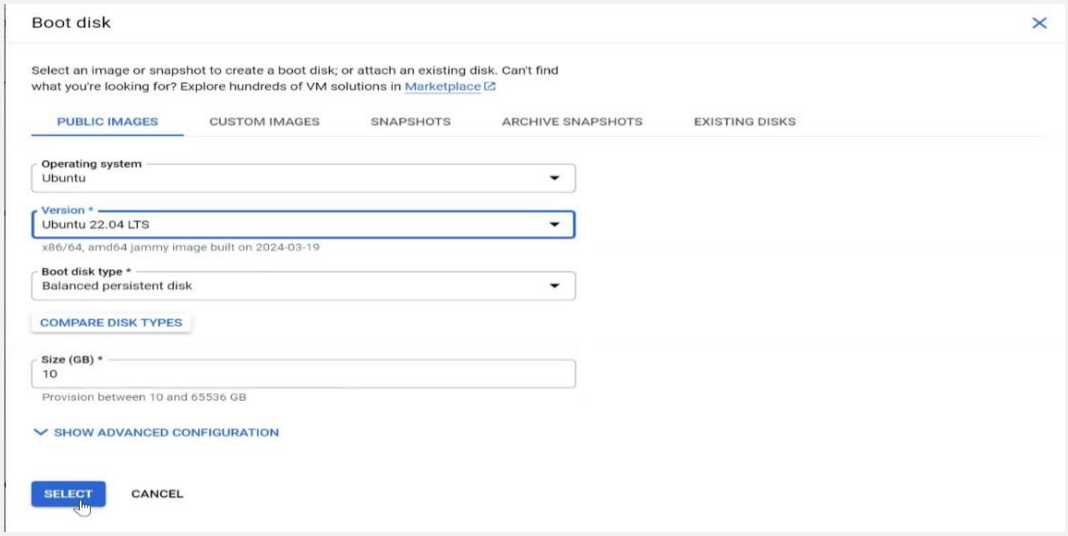


- Step 2:
  - Now, we are selecting Storage; *with 2 Processor and 8 GB RAM* as this is the free account and it's enough for us.

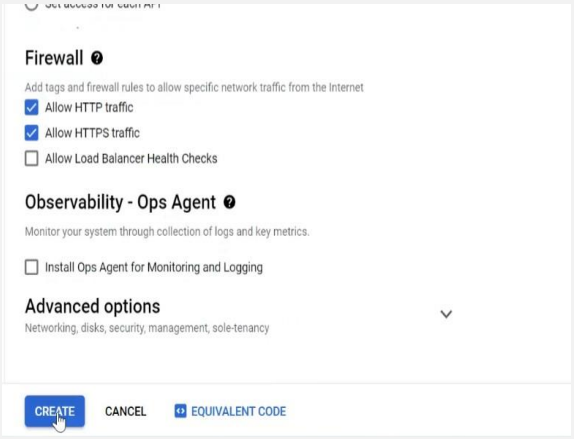


- Step 3:
  - Now, there will be an option of ENABLE DISPLAY DEVICE if you want to record and this VM so Check on that.

- Now, here you can see the option of Boot Disk Click on that from this we can customize our OS. Here, we'll select Ubuntu 22.04 latest version of OS, Disk type will be persistent disk and Disk size we're getting 10 by default but you can increase that later on.



- Step 4:
    - Now, for the remote access and website access we need to allow HTTP & HTTPS traffic. So, Check on HTTP & HTTPS traffic allowance and Click on Create Option.



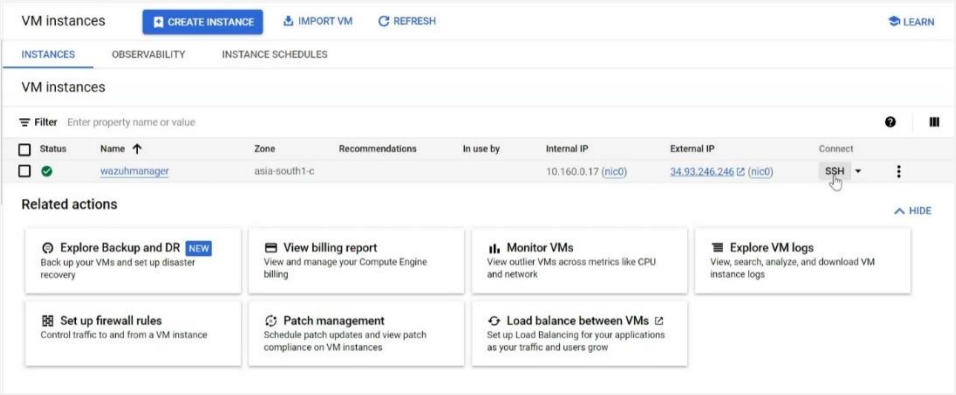
- After Creation of Virtual Machine, we need to modify our Firewall default rules as well. So, for that go to the Firewall.
- In the Firewall rules, edit default-allow-http rule , In the section of ports and protocols select the specific ports and Add 80 & 9000 for http.

default-allow-http	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:80, 9000	Allow	1000	default	Off
default-allow-https	Ingress	https-	IP ranges: 0.0.0.0/0	tcp:443, 1514-1515	Allow	1000	default	Off
inbound	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:1514-1515	Allow	1000	default	Off

- In the rule of default-allow-https add 443 & 1514-1515 for https.

- Step 5:

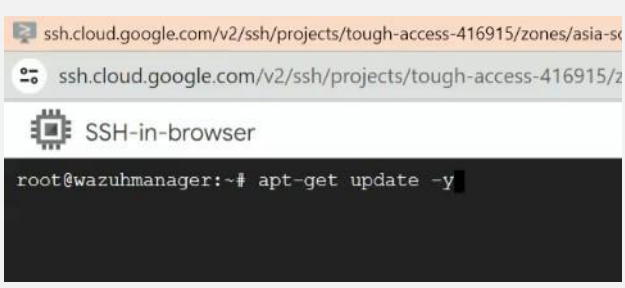
  - Here, we can see we successfully created our VM for WazuhManager.

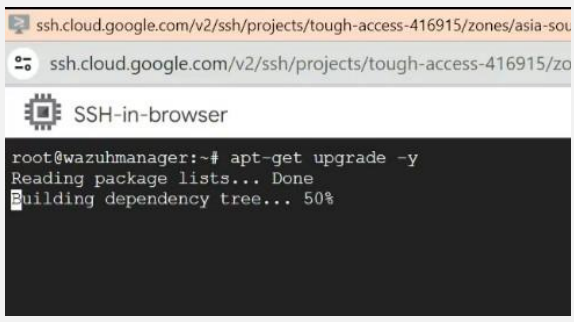


- Now, follow the below steps for installation of Wazuh in the VM.

  - Step 1:

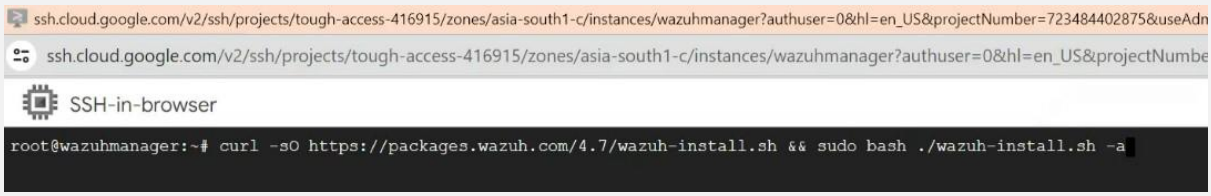
    - Firstly, Take the SSH access of your VM.
    - After getting SSH you'll see there's a name of your GCP username and @VM name from that switch to the root user using command : sudo su –
    - Now, Update all the existing packages and install the newer version of that using the apt-get update & upgrade command.



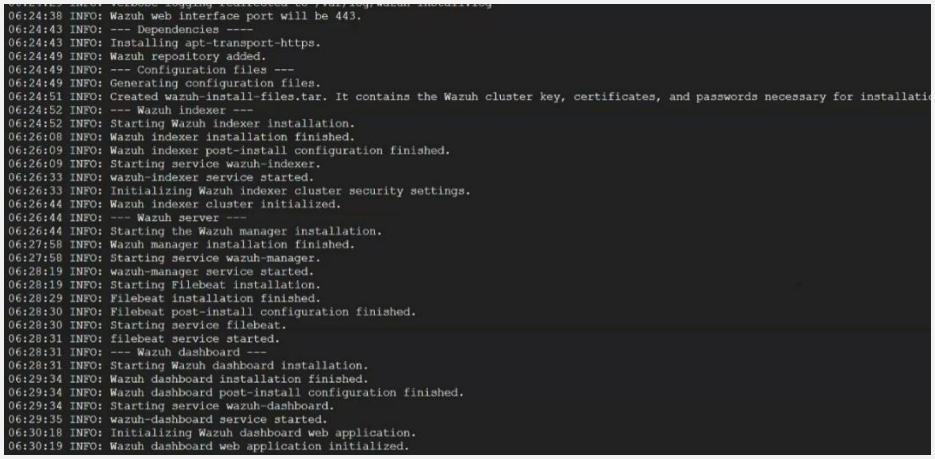


- Now, we are installing the Wazuh using the website URL which we get from the Wazuh official website documentation.

  curl -sO https://packages.wazuh.com/4.7/wazuh-install.sh && sudo bash ./wazuh-install.sh -a

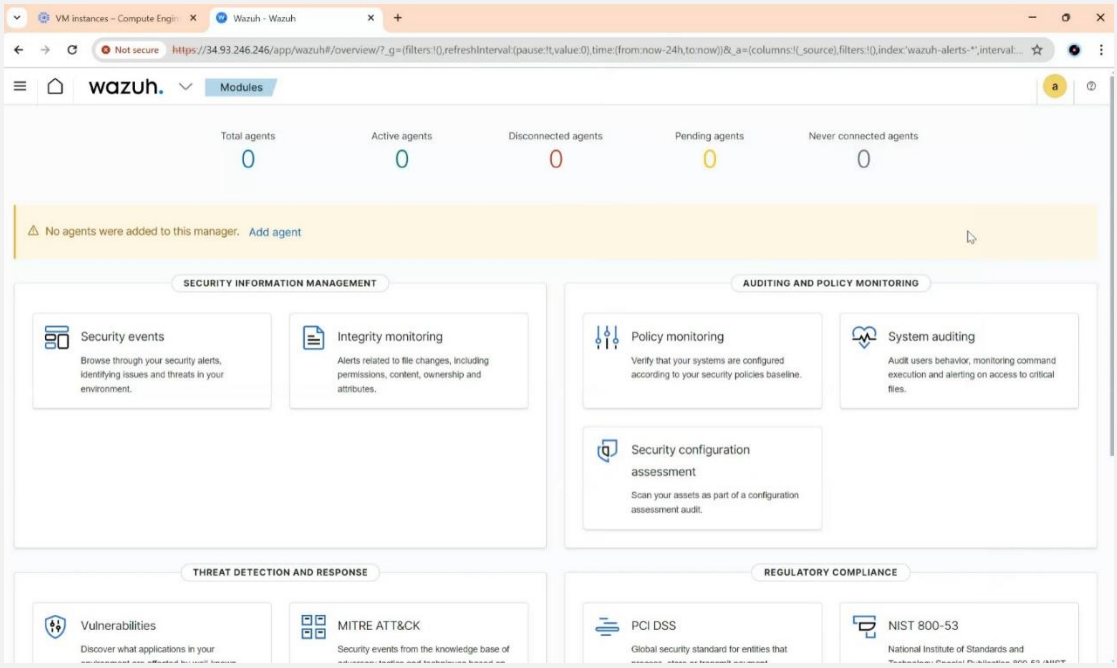


- You can see here our installation begin and first it will install dependencies, Configuration files and then Wazuh Indexer, Wazuh Server and Wazuh Dashboard and the summary of it.



After the Complete Installation you can see there's a username and password.

- Now, Copy that username password. Now go to website search for https://YourwazuhmanagerPublicIPAddress:443 and you'll get security error click on Advance there'll be a login page of Wazuh and paste it here that Username password.
- After a while you can see the Wazuh Dashboard that means you successfully installed Wazuh on your system.



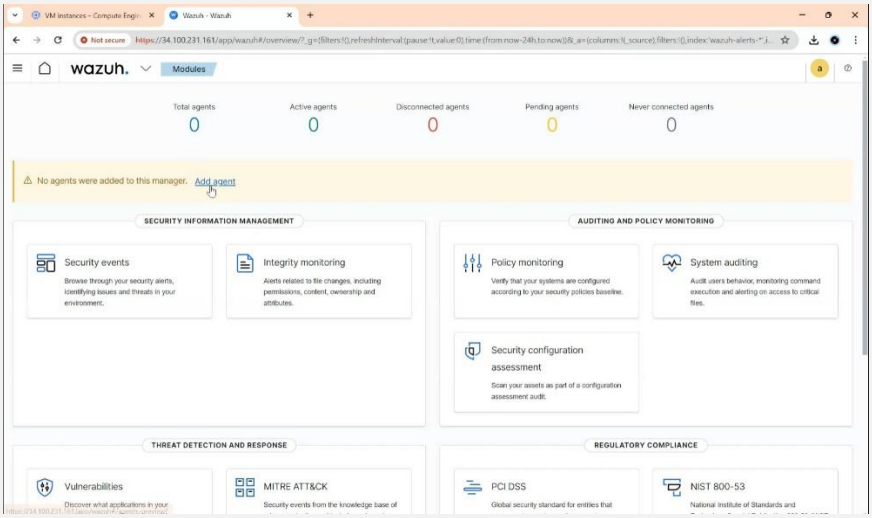
3.2 Adding Agent in Wazuh

- Once the Wazuh manager is installed and operational, proceed to add agents to monitor target Windows systems. Configure the agent to communicate with the Wazuh manager by specifying its IP address or hostname. Verify the agent connectivity by checking the Wazuh manager console for agent registration and status.

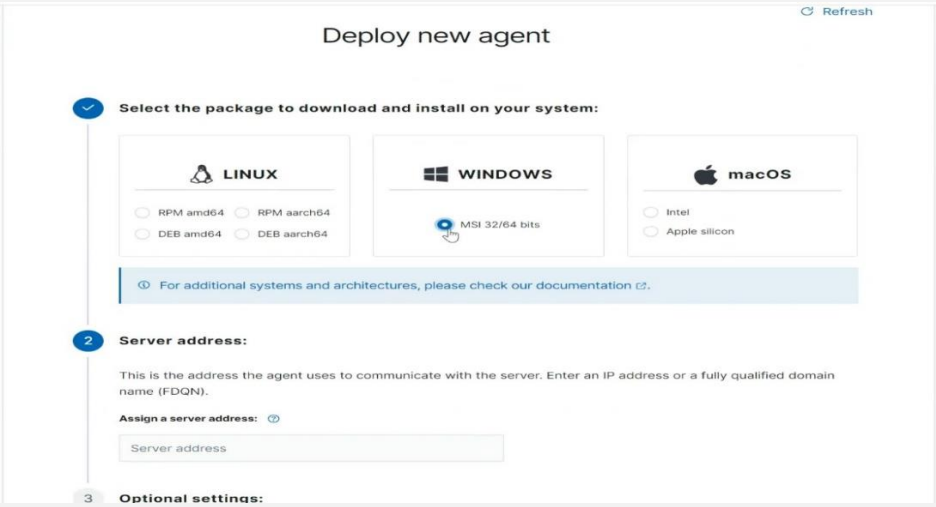
- Before adding the agent, make sure you have a Windows 10 virtual machine (VM) installed, either on your local machine, another host, or a cloud platform, with internet connectivity, and assign it an IP address.

- Once Windows 10 is installed and connected to the internet, proceed with the following steps:

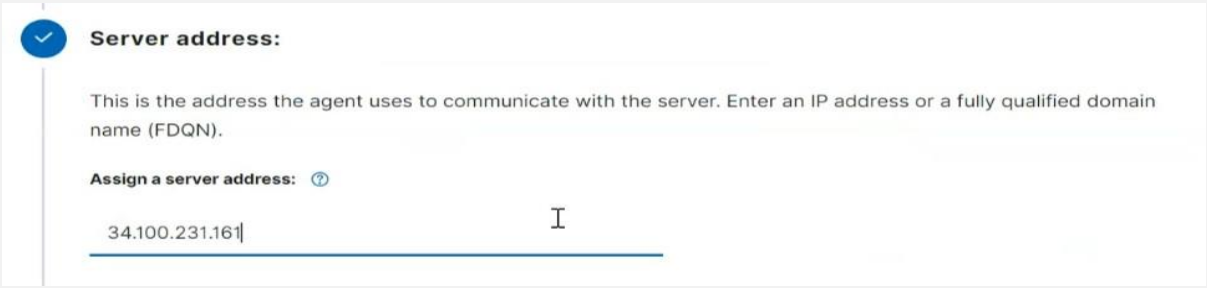
  - Step 1:
    - Now, After the Successful installation of Wazuh we see the dashboard view go to that Wazuh and at there you'll see Add Agent Option select that.



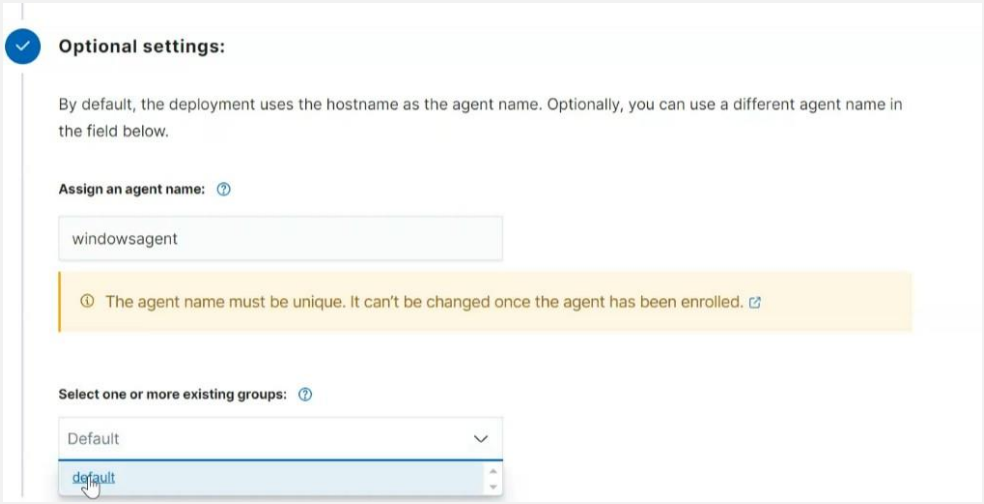
- Step 2:
  - Now, deploy a New Agent in that you have to select the OS like which OS you're adding as an Agent here we're adding Windows so we'll select Windows.



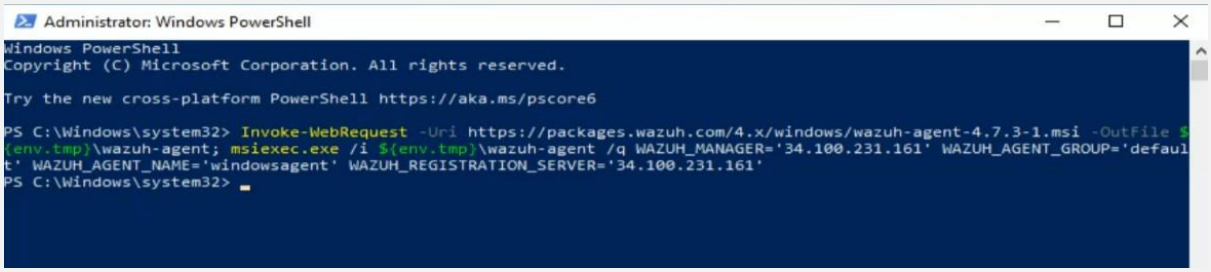
- Step 3:
  - Now, we'll give the Server IP Address which'll be the Public IP Address of our Wazuh.



- Step 4:
    - After giving the IP, we'll assign a name of our agent here I'm giving windowsagent. I'm adding this agent in existing agent group which is default.

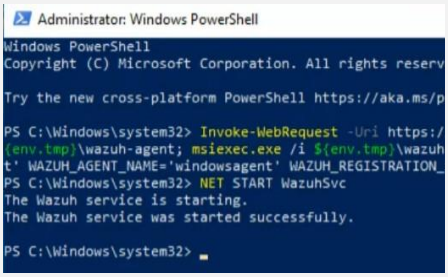


- Step 5:
    - Now, run the following command in your agent. Start Windows PowerShell as an administrator and copy paste this command. It will authenticate the agent and check the connectivity.

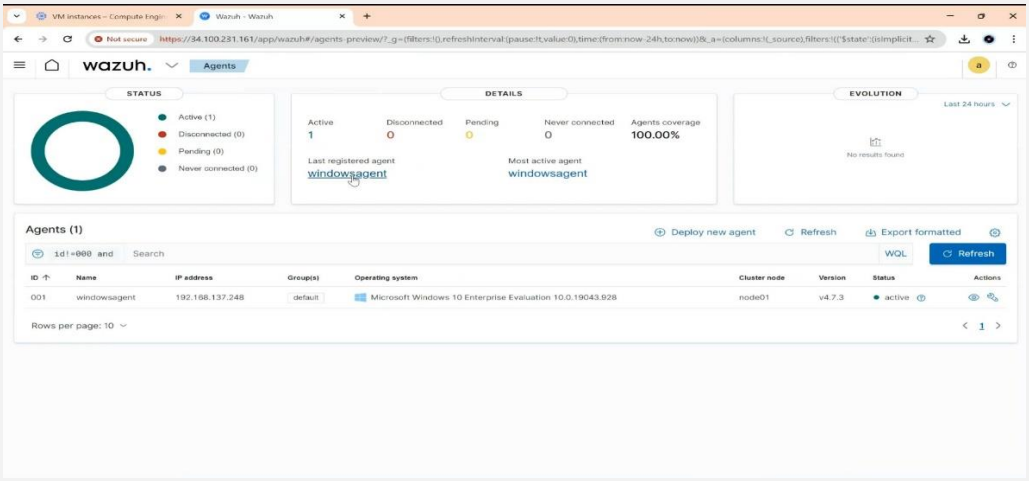


- Step 6:

- Once you run that command after that you need to start the Wazuh Service for that you need to run this command : NET START WazuhSvc



- After the Successfully running the commands in Windows 10 machine which is your Agent now. To see that agent in Wazuh Refresh or Reload the Wazuh and then you can see there's an active agent named windowsagent.



- Step 7:

- Now if you want to check this through the command line or else you want to add agent through CLI Based then take SSH of wazuhmanager and go to the /var/ossec/bin/ at there you'll find the manage-agents file go into that file and from here you can edit and modify your agent.

```
root@wazuhmanager:~# cd /var/ossec/
.ssh/            api/            etc/            lib/            ruleset/        var/
active-response/ backup/         framework/      logs/           stats/          wodles/
agentless/       bin/            integrations/   queue/          tmp/
root@wazuhmanager:~# cd /var/ossec/bin/
root@wazuhmanager:/var/ossec/bin# ls
agent_control    manage_agents      wazuh-apid      wazuh-db           wazuh-logtest         wazuh-regex
agent_groups     rbac_control       wazuh-authd     wazuh-dbd          wazuh-logtest-legacy  wazuh-remoted
agent_upgrade    verify-agent-conf  wazuh-clusterd  wazuh-execd        wazuh-maild           wazuh-reportd
clear_stats      wazuh-agentlessd   wazuh-control   wazuh-integratord  wazuh-modulesd        wazuh-syscheckd
cluster_control  wazuh-analysisd    wazuh-csyslogd  wazuh-logcollector  wazuh-monitord
root@wazuhmanager:/var/ossec/bin# ./manage_agents
```

- Here you can see it's showing our windowsagent.

```
root@wazuhmanager:/var/ossec/bin# ./manage_agents


****************************************
* Wazuh v4.7.3 Agent manager.          *
* The following options are available: *
****************************************
   (A)dd an agent (A).
   (E)xtract key for an agent (E).
   (L)ist already added agents (L).
   (R)emove an agent (R).
   (Q)uit.
Choose your action: A,E,L,R or Q: L

Available agents:
   ID: 001, Name: windowsagent, IP: any

** Press ENTER to return to the main menu.
```

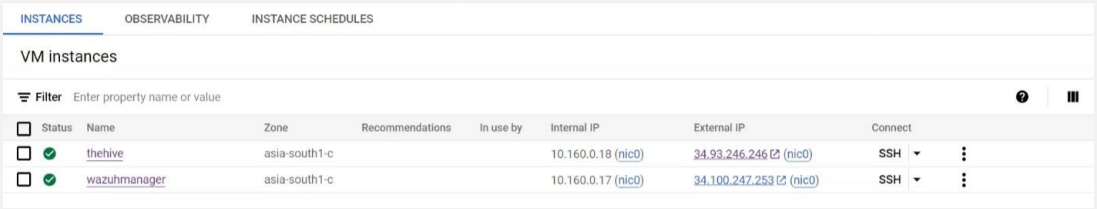
4. Setting Up the TheHive

- TheHive is a powerful Security Orchestration, Automation, and Response (SOAR) platform designed to facilitate incident response and threat intelligence activities within organizations. It serves as a centralized hub for managing security incidents, enabling security teams to streamline workflows, automate response actions, and collaborate effectively.

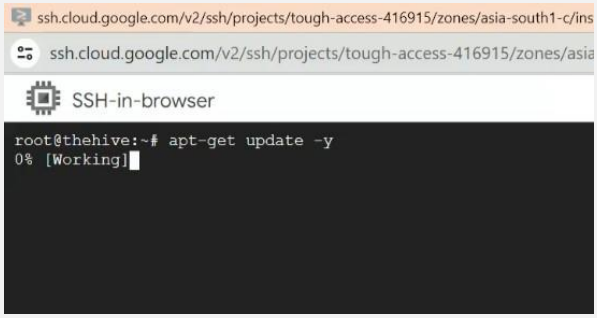
- Now, let's proceed with the installation of TheHive on Google Cloud Platform (GCP).

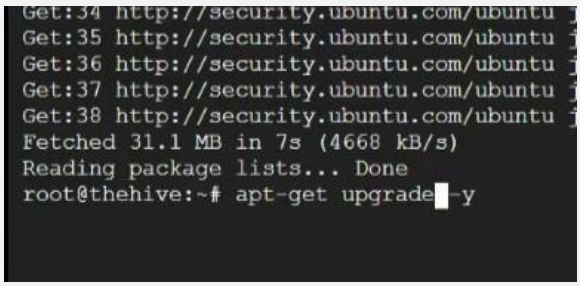
4.1 Installation of TheHive

- Step 1:
  - For TheHive installation, we'll create another VM, similar to how we did for Wazuh, this time on Google Cloud Platform (GCP) as I'm performing its installation on GCP.

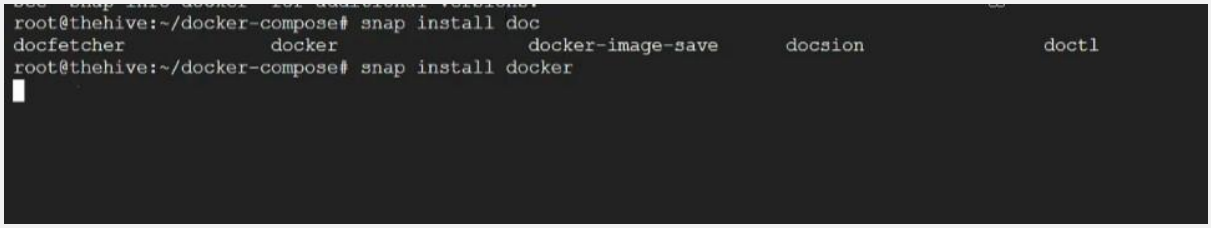


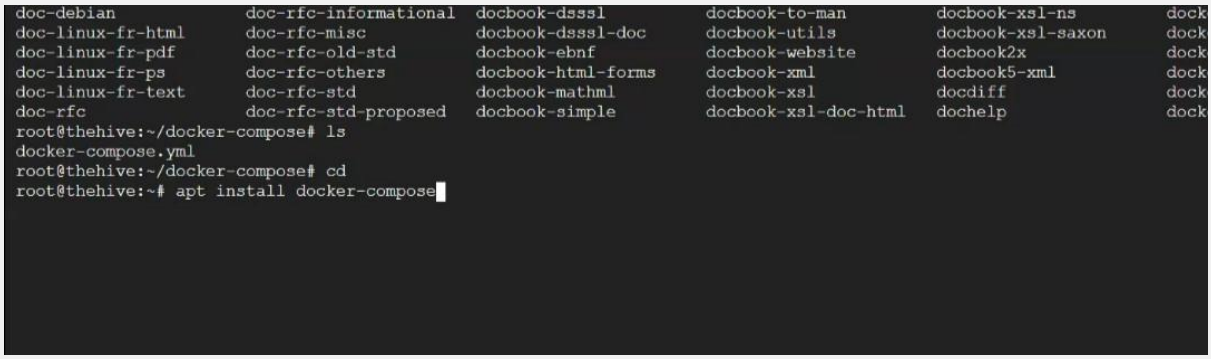
  - Firstly, Now took the SSH of TheHive and Update the existing packages and Indexes for that do apt-get update | apt-upgrade -y.



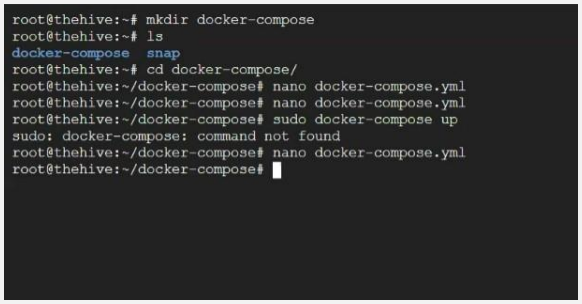


- Step 2:
  - Now, I'm installing TheHive from the GitHub source which I'll add at the end in the references.
  - Here, we're installing TheHive using docker so for that we need to install docker in our VM so do snap install docker and apt install docker-compose.

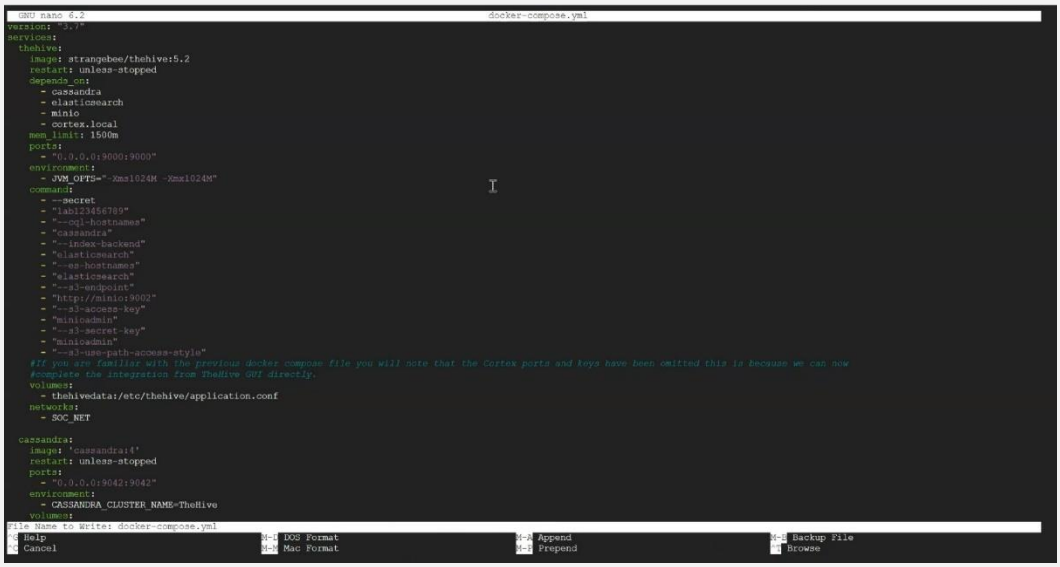




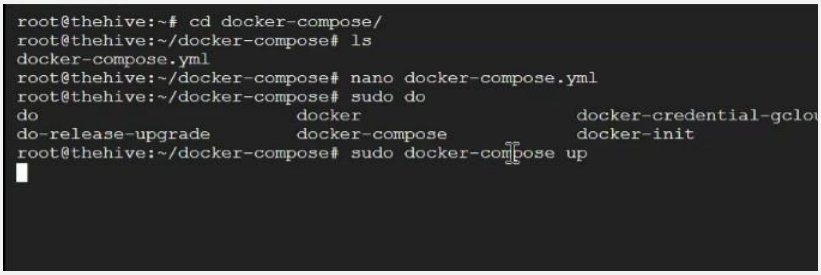
- Step 3:
  - After installing docker and docker-compose we'll create a folder name docker-compose and under that we'll create a nano file named docker-compose.yml



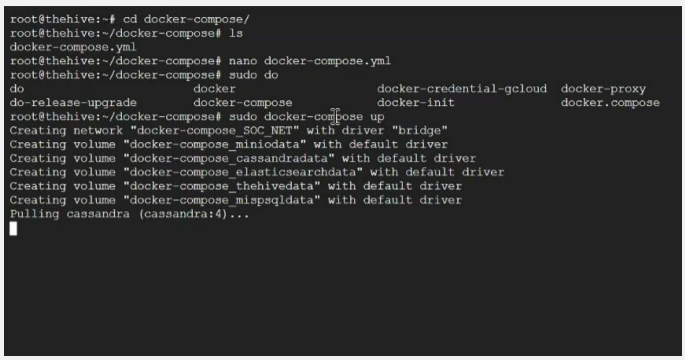
  - In that we'll add all the dependencies and also the repositories which we use for the installation of TheHive.

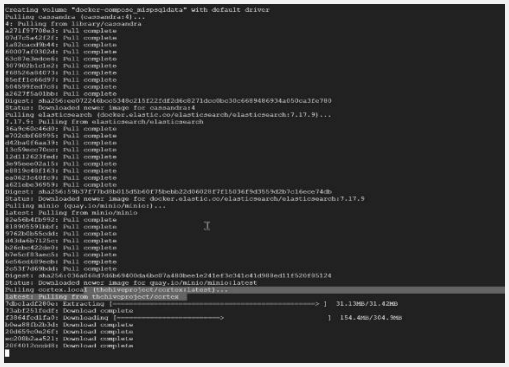


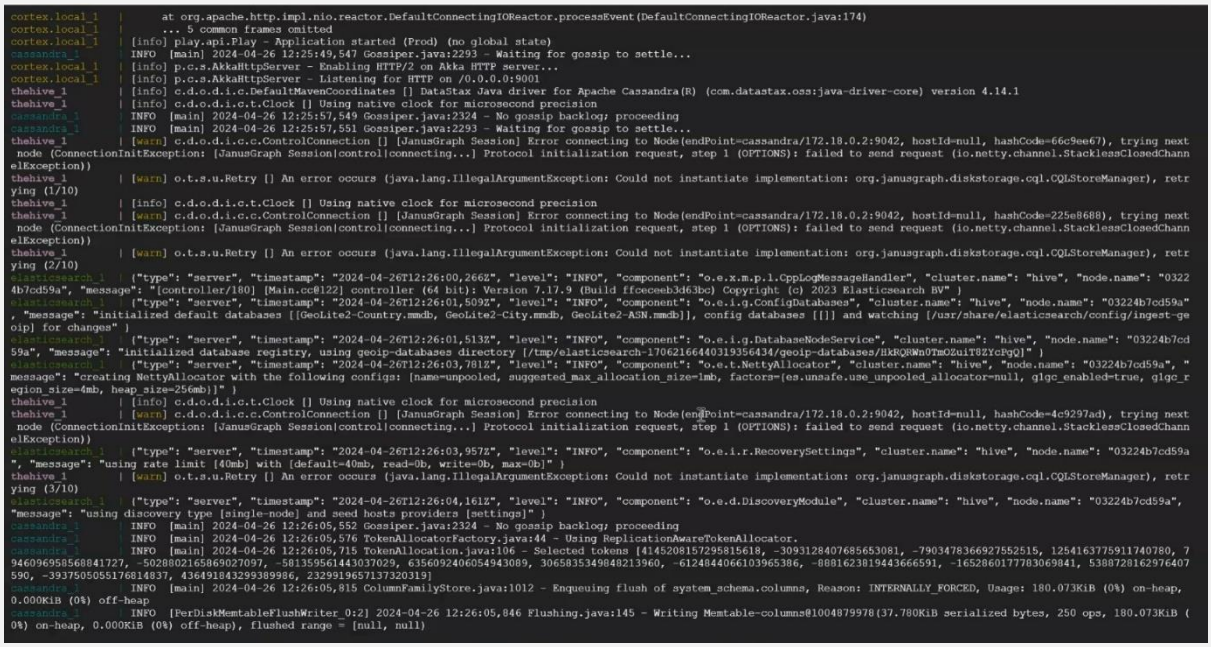
- Step 4:
  - We created a nano file named docker-compose.yml Now, we'll start the installation of TheHive. For that we'll run the command *docker-compose up*. Here, you can see the process has been started.



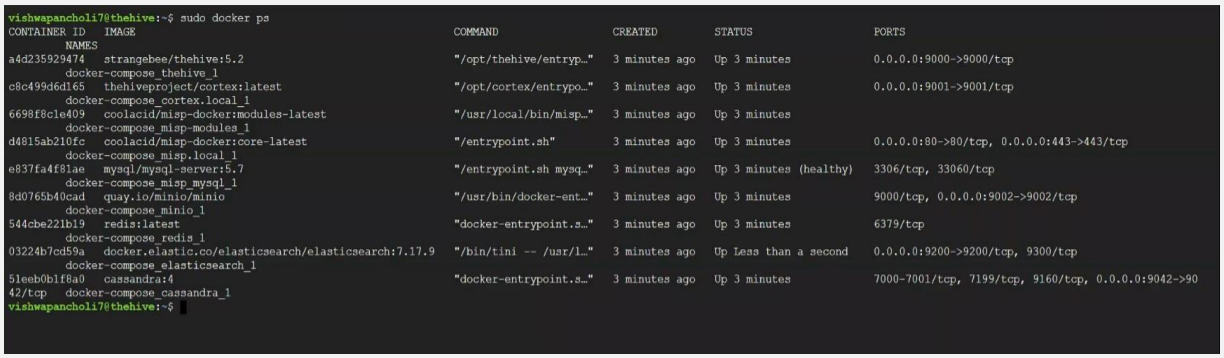
- Step 5:
  - After sometime you can there were many processes are happening for Cortex, TheHive, MISP, Elastic search and many more.



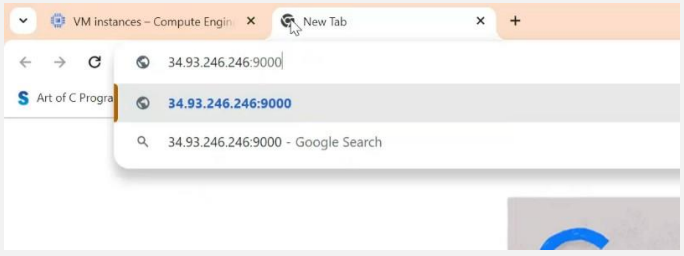




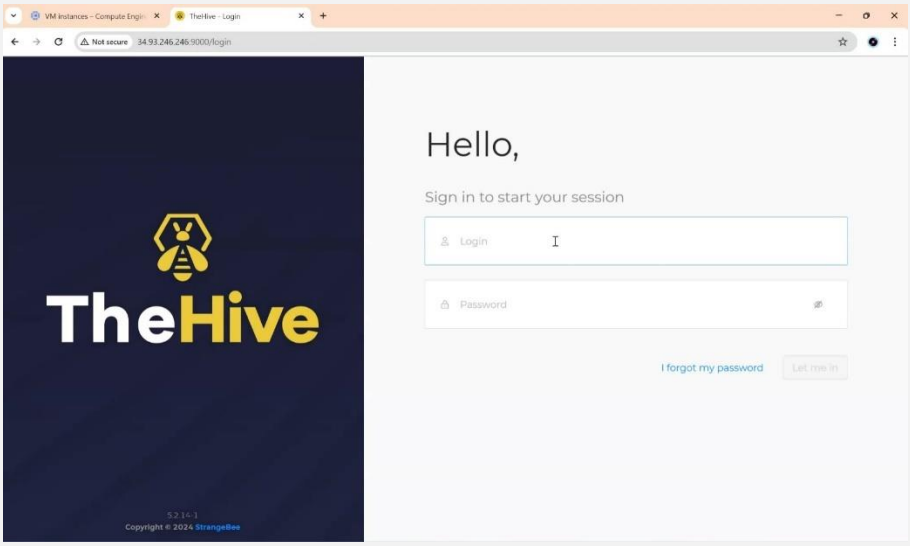
- After some time take a new SSH but don't shutdown the existing SSH processes. In the new SSH switch to the root user and run this command here you can see our TheHive is successfully running on the port number 9000.



- Step 6:
  - Now, go to your website and copy paste your TheHive's Public IP Address with port number 9000 like this YourIPAddress:9000



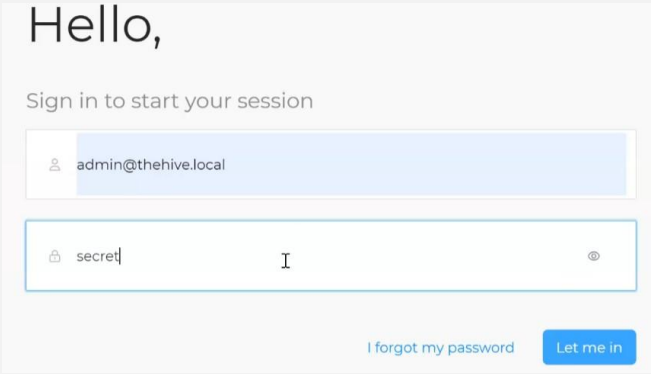
- After this you'll see a login page



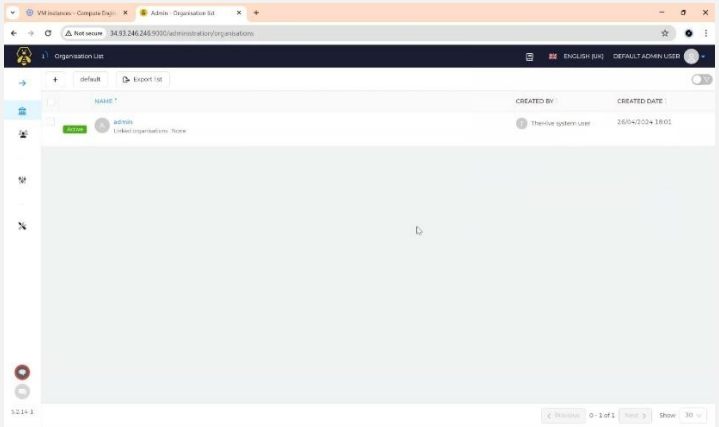
- Step 7:
  - Here, enters the default username password.

    Username: admin@thehive.local

    Password: secret



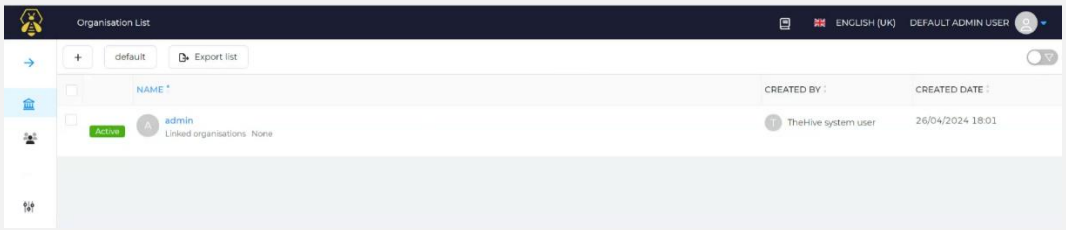
  - After the logging in TheHive you can see the Console View of it.

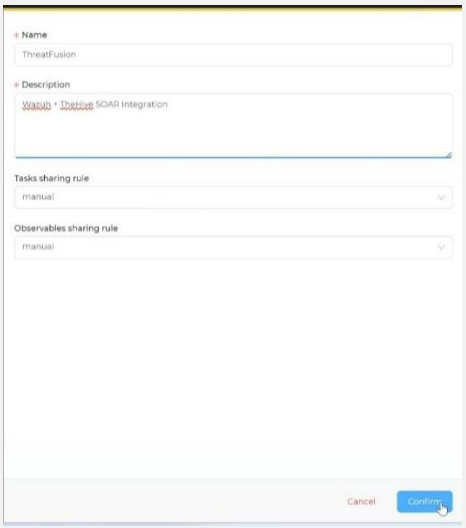


4.2 Organization and User Management

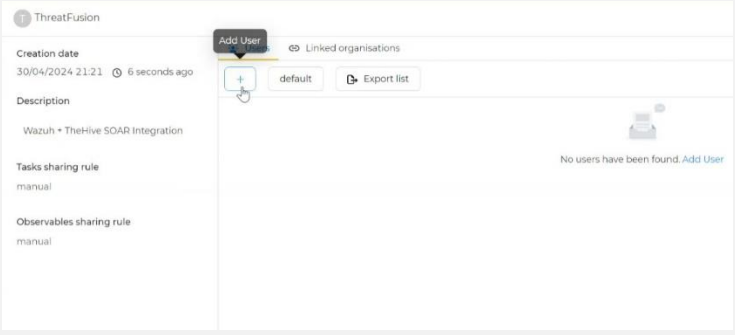
- In this section, we'll set up user management by creating users and organizations within TheHive:

- Step 1:
  - In your console you can see the organisation list there's a default admin organisation we'll add our own organisation named ThreatFusion.



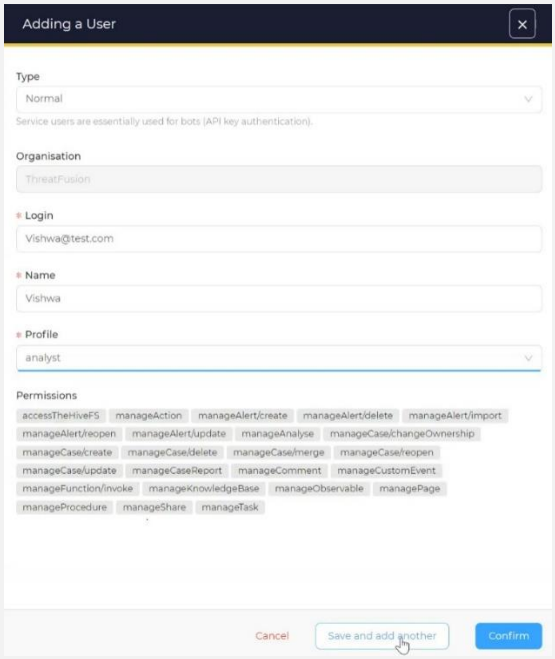


- Step 2:
  - Now click on that organisation there's no user in it so we'll add the user in it and for Click on Add User.

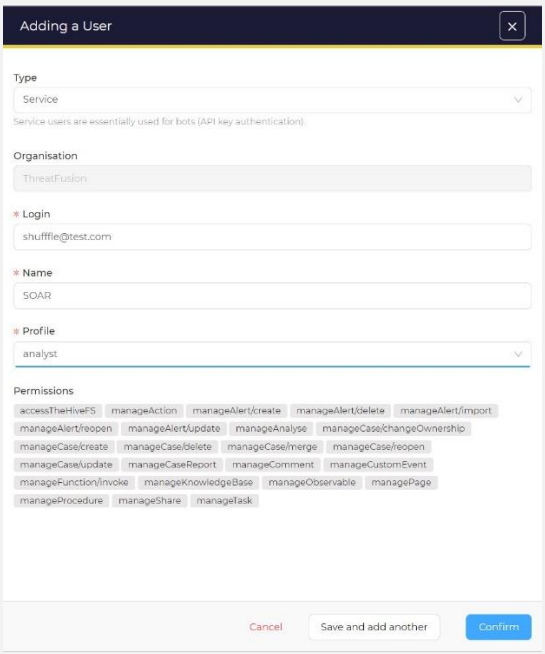


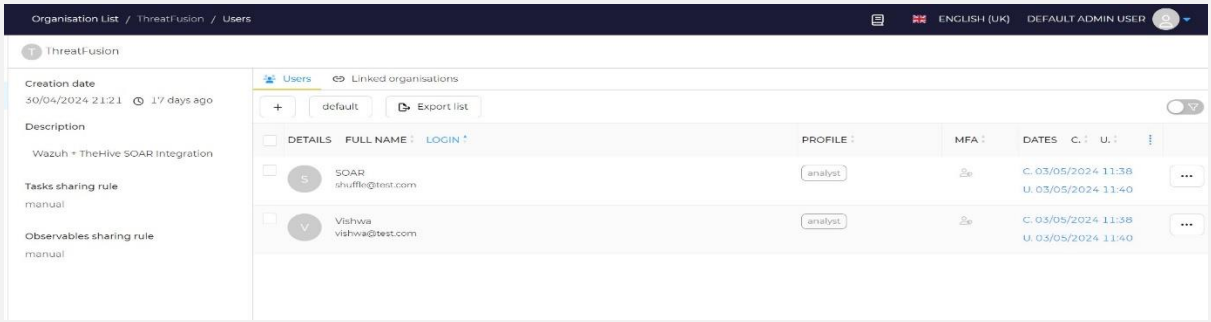
- Here we get 2 types of Users 1st is normal and 2nd one is Service so we'll create 2 users.

1st User: Vishwa



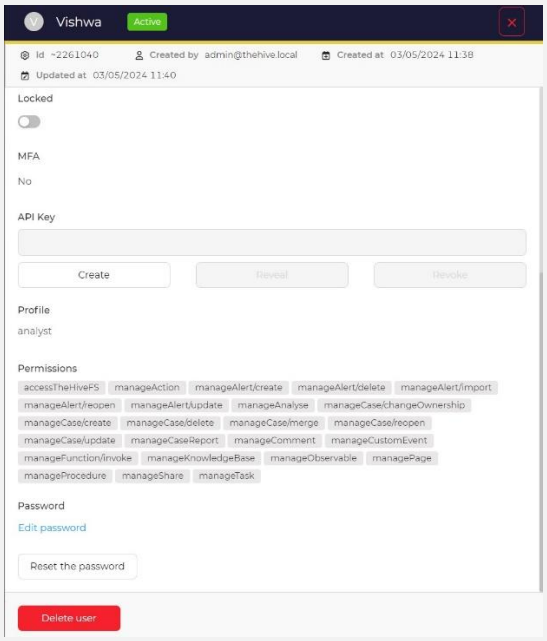
2nd User : SOAR

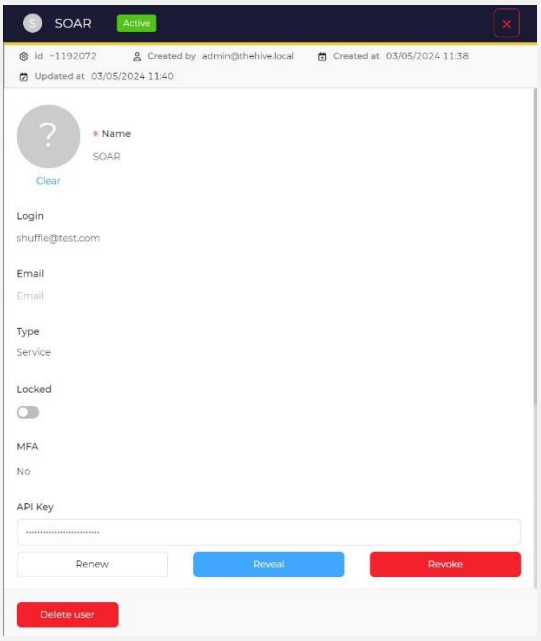




- Step 3:
  - Now, in the ThreatFusion User list you can see Vishwa and SOAR users. We'll edit this user by adding a password and API Key.

- We'll add Password for our normal user and for the Service user we'll generate the API Key. Now remember this API Key we'll use as in our workflow which we'll do in Shuffle.





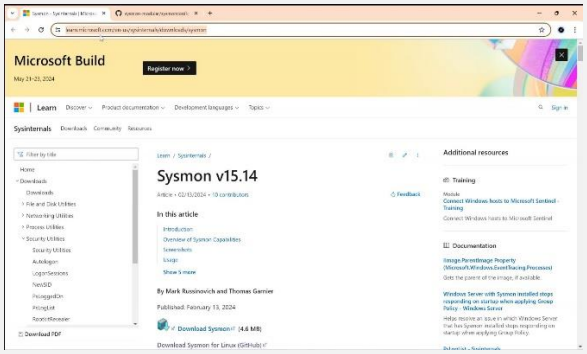
5. Telemetry Setup on Wazuh Dashboard

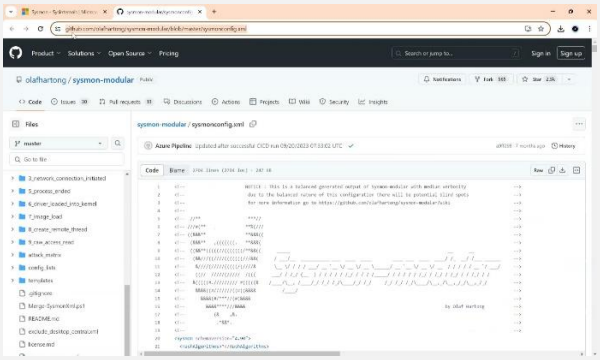
- In this section, we'll integrate the Sysmon tool and the Mimikatz tool into our Windows agents. Subsequently, we'll observe Mimikatz alerts within Wazuh. Here's a brief description of Sysmon and Mimikatz:
- Sysmon (System Monitor) :
  - Sysmon is a Windows system service and device driver that monitors and logs system activity to the Windows event log. It provides detailed information about process creations, network connections, and changes to file creation time. Sysmon enhances visibility into system activity, aiding in threat detection and incident response by capturing valuable forensic data.
- Mimikatz :
  - Mimikatz is a powerful post-exploitation tool that allows users to extract credentials from memory, perform pass-the-hash attacks, and perform other credential manipulation tasks. It is commonly used by attackers to escalate privileges and move laterally within a compromised network. Detecting Mimikatz activity is crucial for identifying potential security breaches and responding to them effectively.

5.1 Integrating Sysmon and Mimikatz

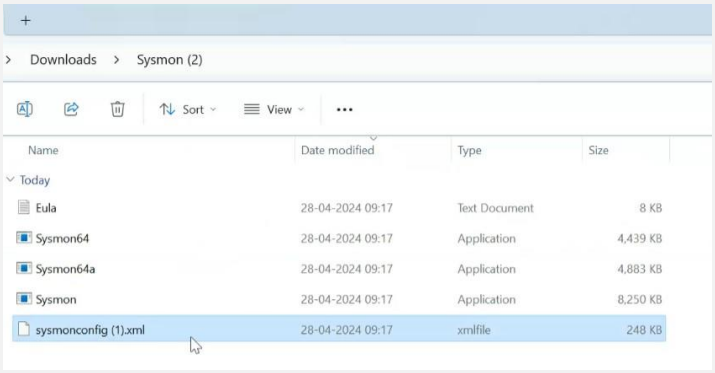
- Now, let's proceed with adding Sysmon and Mimikatz to our Windows agents and configuring Wazuh to detect Mimikatz alerts.

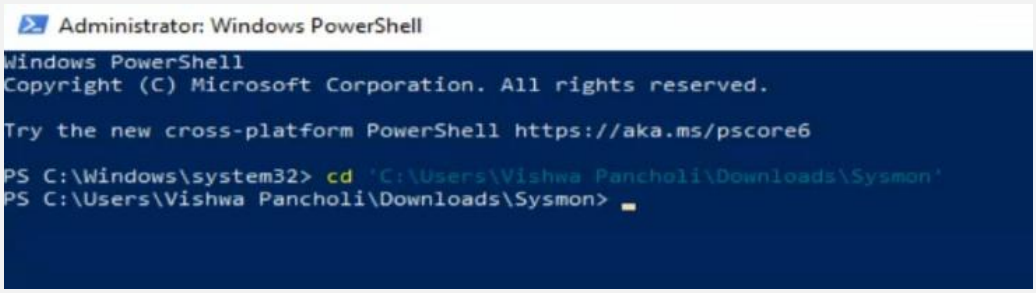
  - Step 1:
    - We are generating the alert of Mimikatz in Wazuh so for that firstly we'll install the sysmon in our windows agent for that I took the GitHub source so from that i download the file of sysmon configuration and also download the sysmon tool zip from its official website and as well extract that folder and configuration file in it.

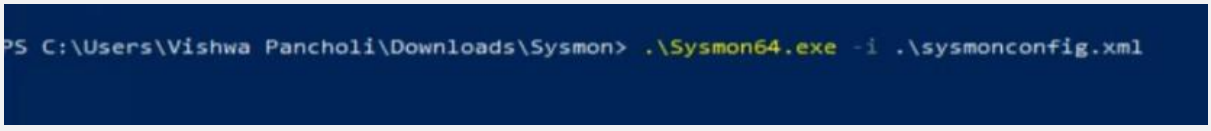


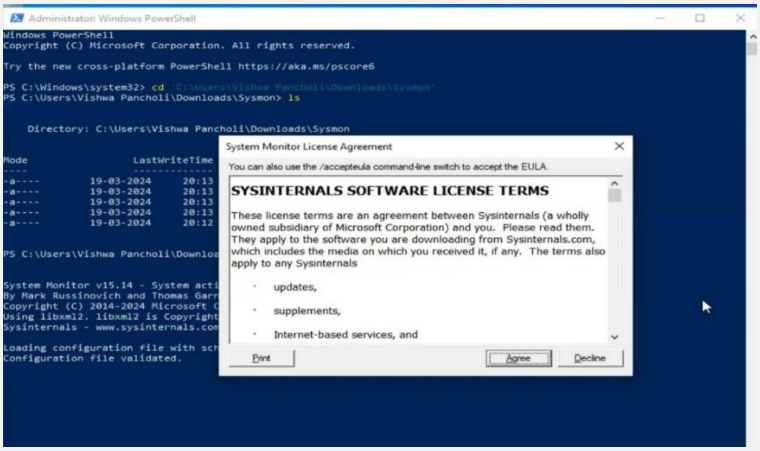


- Step 2:
  - Now, we download this sysmon for that start the PowerShell as an administrator and go into that Directory where you installed folder copy paste that path now, we'll run this command : [Folder name] -i [configurationfile]





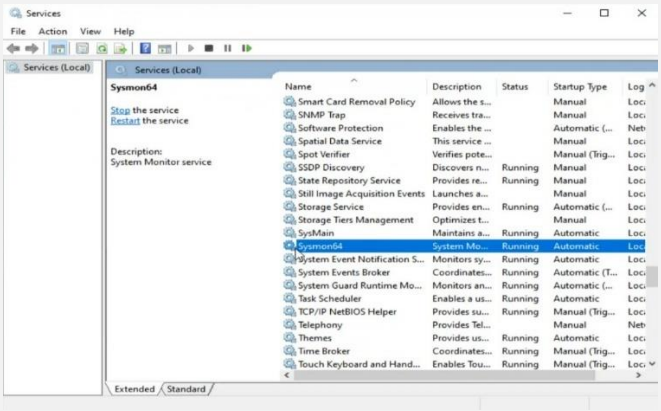


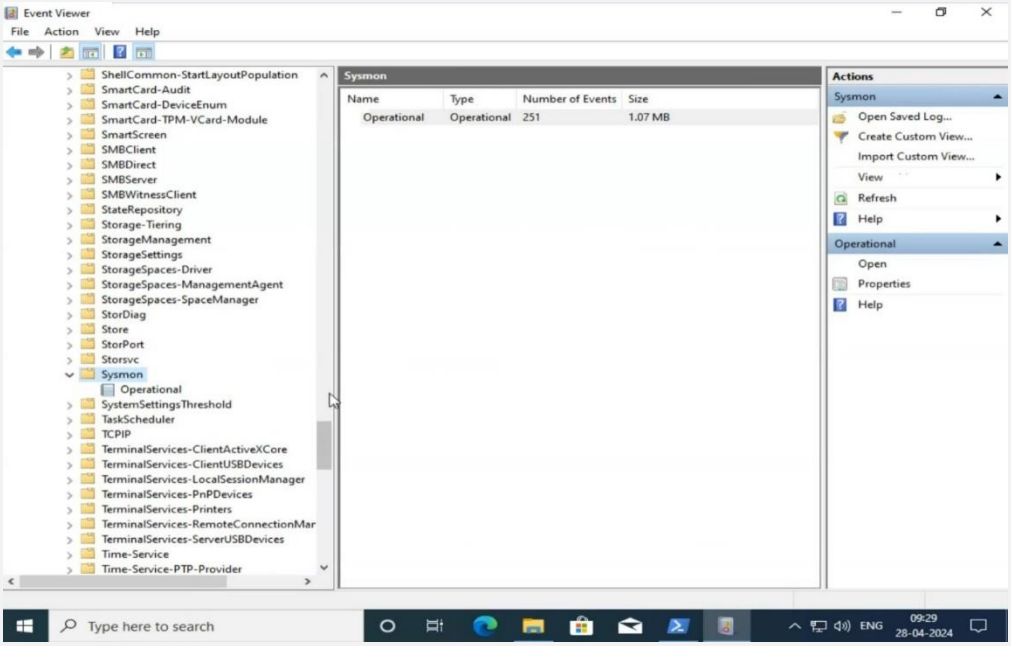


- You can see there's a popup for the Agree Terms and Conditions and you successfully installed the sysmon.

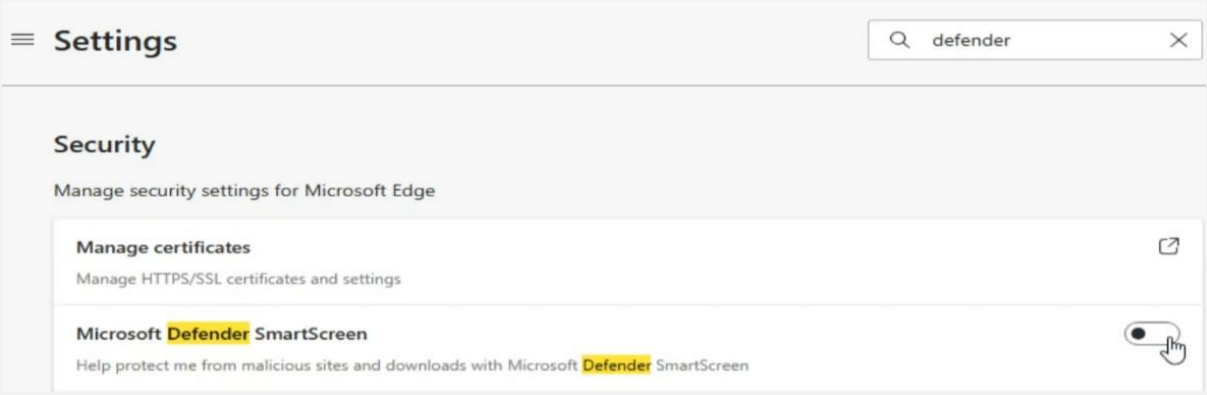
```
PS C:\Users\Vishwa Pancholi\Downloads\Sysmon> .\Sysmon64.exe -i .\sysmonconfig.xml

System Monitor v15.14 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2024 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.90
Configuration file validated.
Sysmon64 installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon64..
Sysmon64 started.
PS C:\Users\Vishwa Pancholi\Downloads\Sysmon>
```

- Step 3:
  - Now you can verify this by going to services there will be a sysmon64 service or else you can go to event viewer -> Applications and Service Logs -> Microsoft -> Windows -> Sysmon

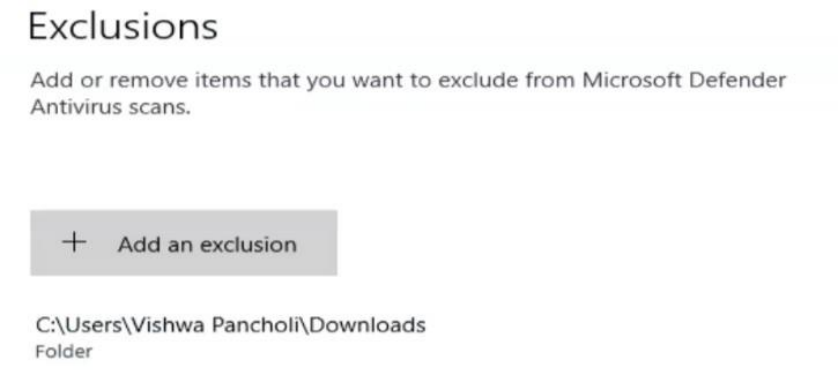




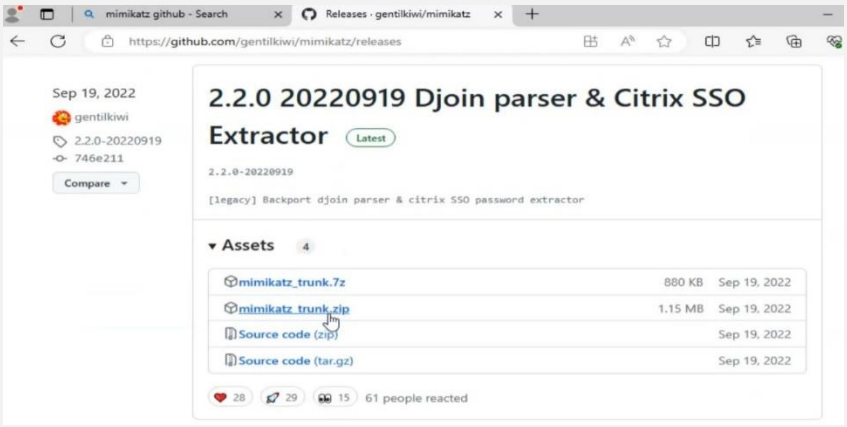
- Step 4:
  - Now we are going to add Mimikatz tool in our windows agent so for that firstly we need to do some security configuration.
  - Firstly, go to your browser -> settings -> Search for Defender -> Microsoft Defender SmartScreen -> disable that option.



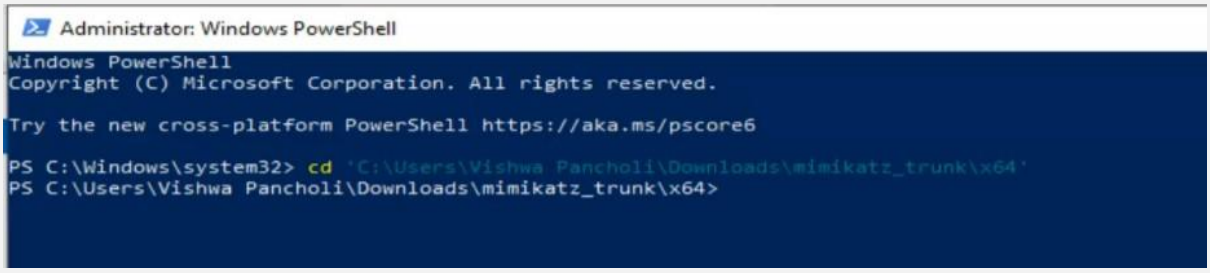
  - After this Go to Windows Security -> Virus & threat Protection -> Under the Virus & threat Protection go to Manage settings -> Exclusion -> Click on Add or Remove -> Click on Add an Exclusion -> Select Folder Option and Select the downloads and done.



- Step 5:
  - Now, I'm downloading the Mimikatz tool from the GitHub source I'll added that link in References and after downloading the file extract that file.



  - Now, go to the Mimikatz file -> x64 and copy the path and paste it in a PowerShell and do the ls you can see all files now run that mimikatz.exe file



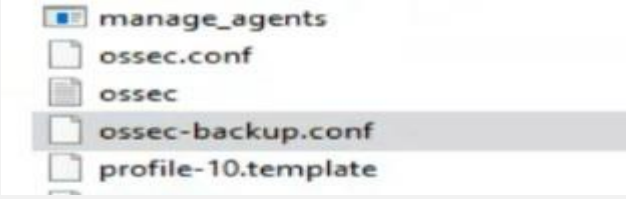
```
PS C:\Windows\system32> cd 'C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64'
PS C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64> ls


    Directory: C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        28-04-2024     11:10          37208 mimidrv.sys
-a----        28-04-2024     11:10        1355264 mimikatz.exe
-a----        28-04-2024     11:10          37376 mimilib.dll
-a----        28-04-2024     11:10          10752 mimispool.dll


PS C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64>
```

- And you can see you enter into the Mimikatz tool.

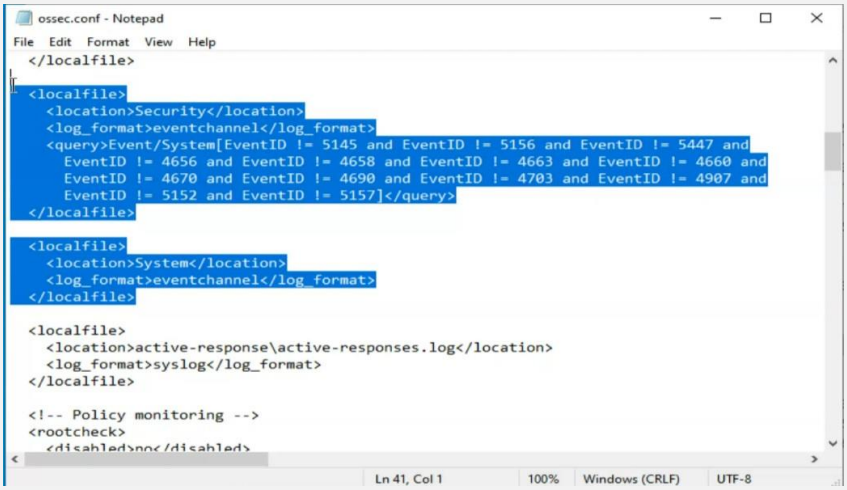
```
PS C:\Windows\system32> cd 'C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64'
PS C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64> ls


    Directory: C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        28-04-2024     11:10          37208 mimidrv.sys
-a----        28-04-2024     11:10        1355264 mimikatz.exe
-a----        28-04-2024     11:10          37376 mimilib.dll
-a----        28-04-2024     11:10          10752 mimispool.dll


PS C:\Users\Vishwa Pancholi\Downloads\mimikatz_trunk\x64> .\mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz #
```

- Step 6:

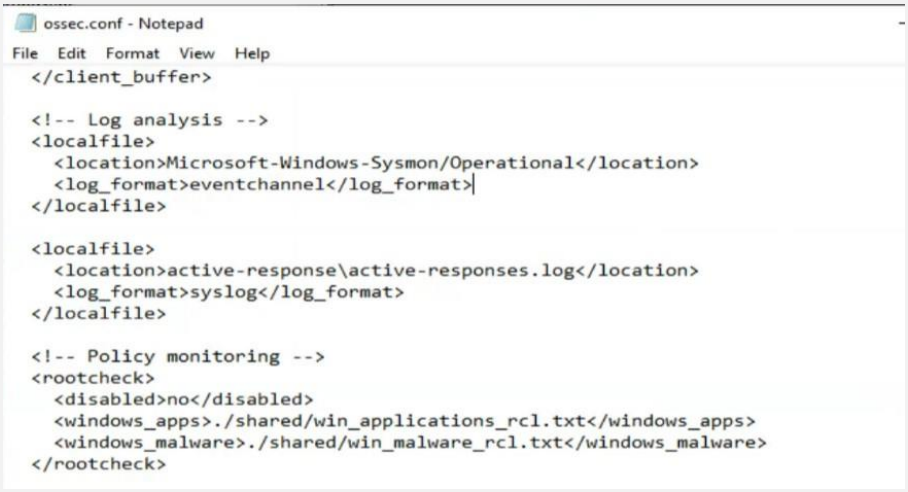
- Now, for generating alerts we need to do integration of Sysmon and Mimikatz first.
- So, for that we'll go to our C drive and there will be a folder named ossec-agent. Go into that folder here you can see the configuration file named ossec.conf we're going to edit that file so before editing that file make a copy of that file named ossec-backup.conf.



- Now Open that original file in notepad as an administrator.
- Now, firstly remove the below configuration :



- And in local file add the Sysmon Full Name from the Event Viewer and edit this as shown below and save this file.

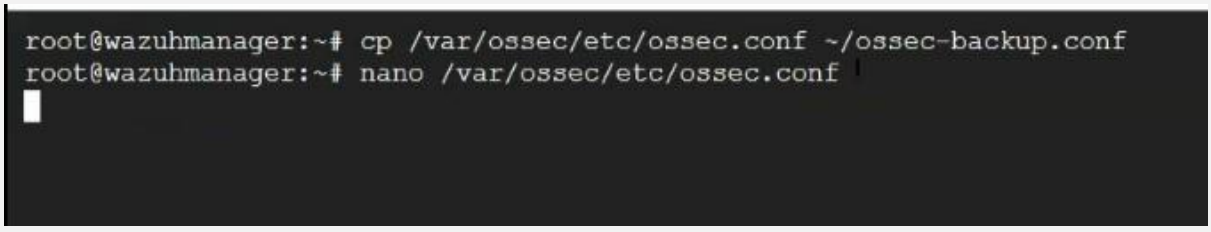


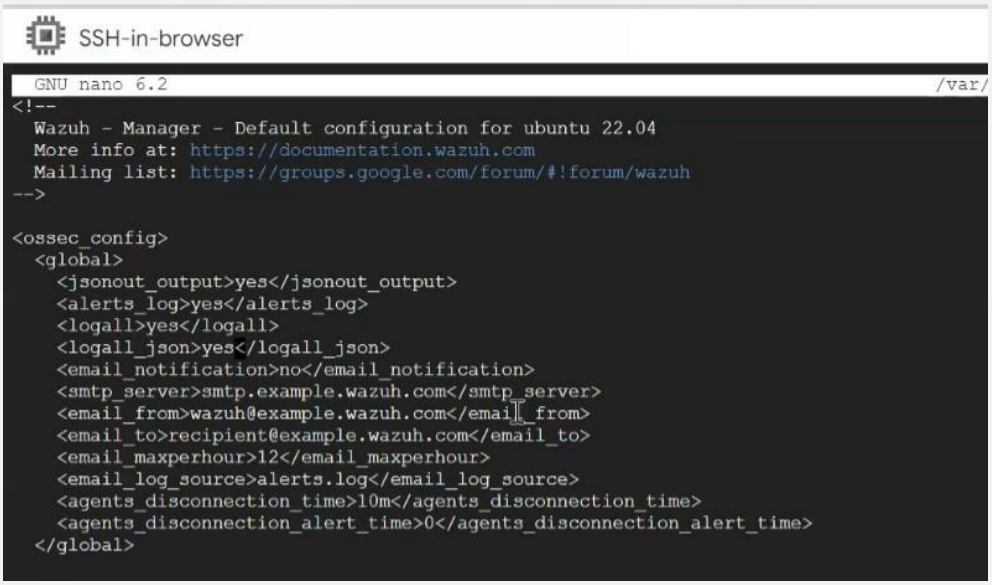
5.2 Alert Detection Configuration in Wazuh

- In this section, we'll generate alerts from the perspective of an attacker, simulating Mimikatz activity. To accomplish this, we'll need to edit rules and configuration files within Wazuh. Let's proceed with making the necessary adjustments:

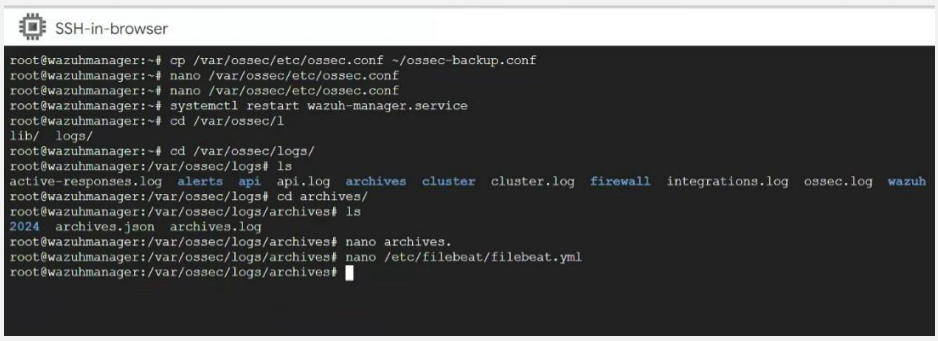
  - Step 1:
- Now, go to the wazuhmanager CLI based and copy the config file as an ossec-backup.conf at a different location.

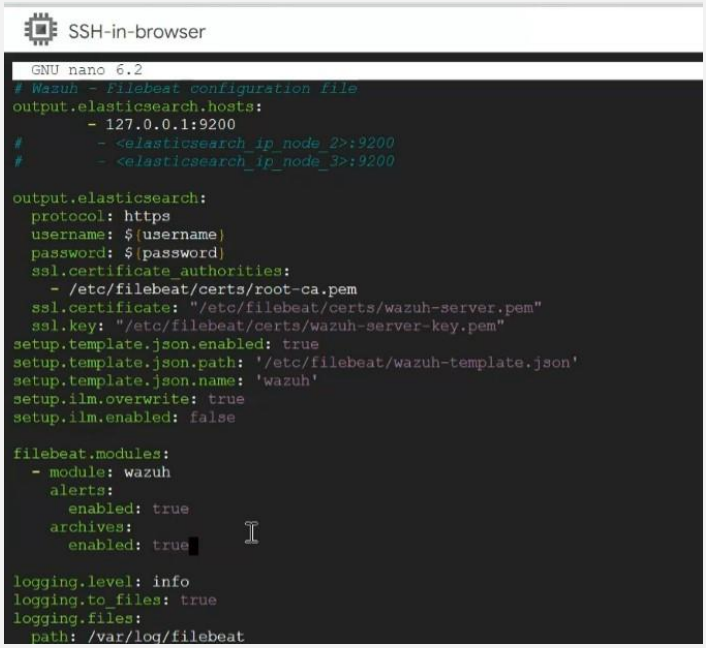
- Now, edit the config file in that <logall>yes</logall> ; <logall_json>yes</logall_json> and restart the wazuh-manager service





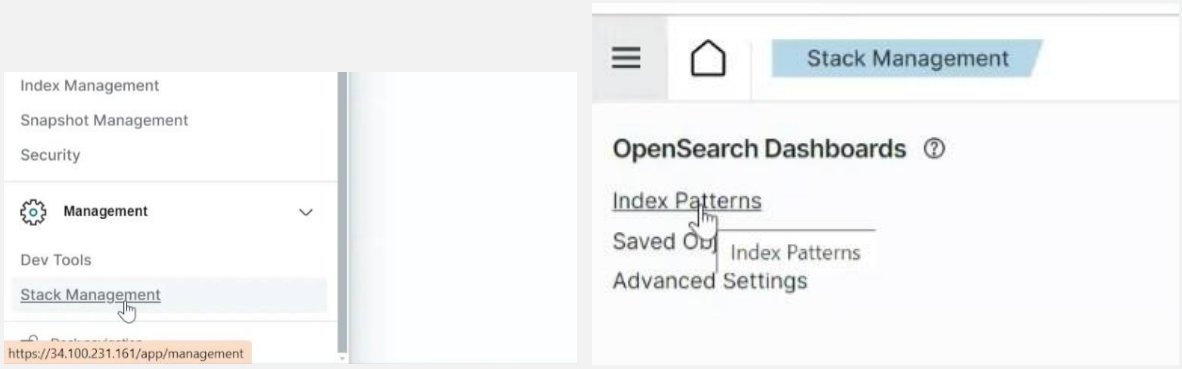
  - Step 2:
- Now, edit the file /etc/filebeat/filebeat.yml in the section of filebeat modules there's an archive enabled make it as a true and save and quit and restart the service of filebeat.

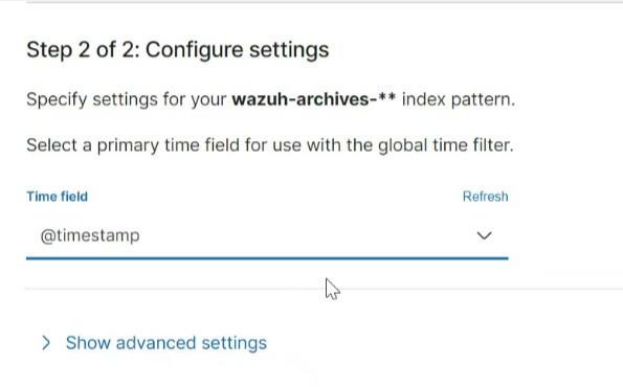


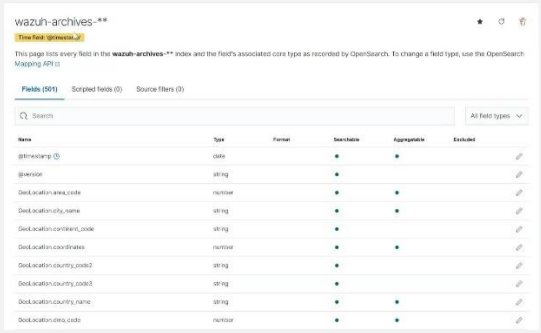


- Step 3:

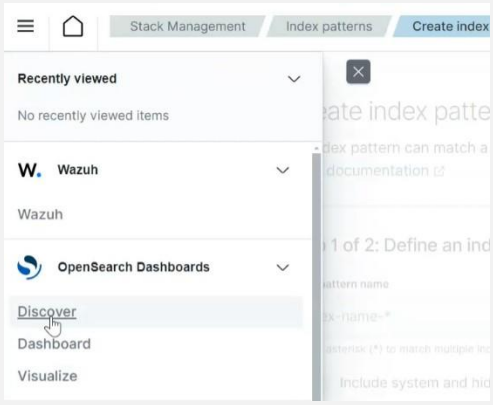
- For the Dashboard Changes go to setting -> Stack Management -> Index Patterns -> Create Index Pattern -> Enter the name : wazuh-archives-** -> Next Step -> Choose @timestamp in time field -> And create.

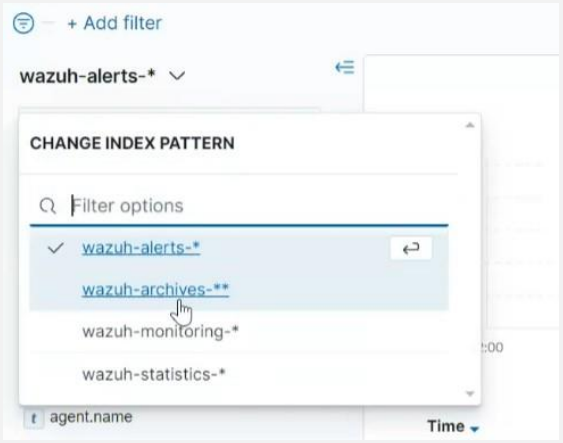




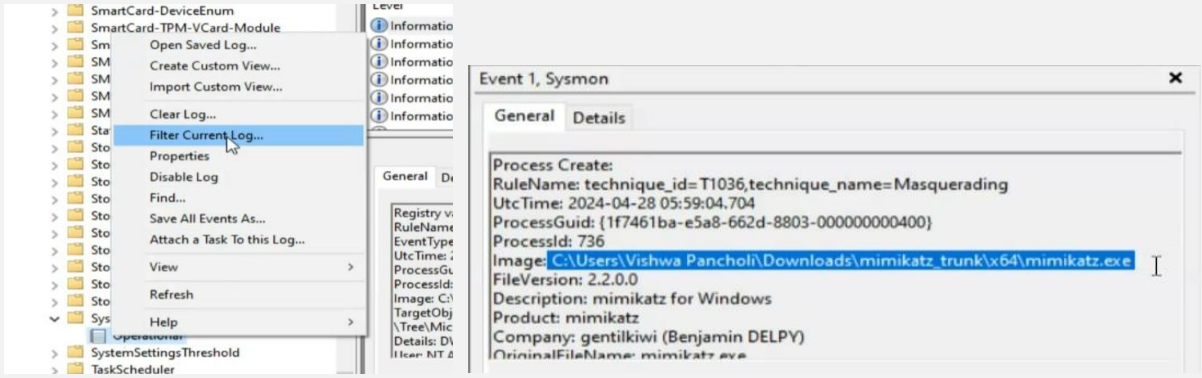


- Here You can see the New Index Pattern.
- Now, Go to Setting -> Discover -> Change Index Pattern -> Select wazuh-archives-**



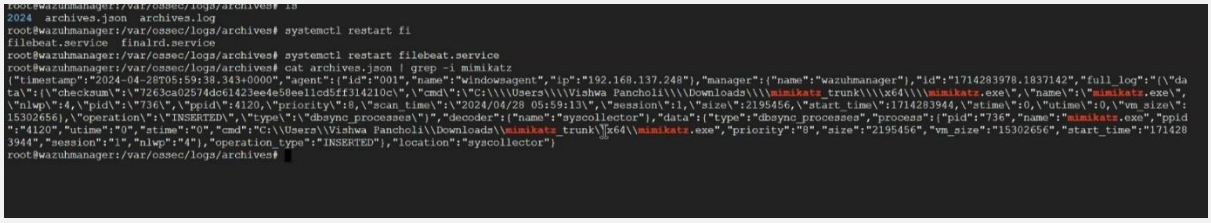


  - Step 4:
- Now, go to your agent and run the Mimikatz.exe again and after that go to event viewer -> Applications and Service Logs -> Microsoft -> Windows -> Sysmon -> right click on Operation -> Select Filter Current Log -> and above task category write Event Id : 1 -> and in the below details you'll find mimikatz path in Image option.

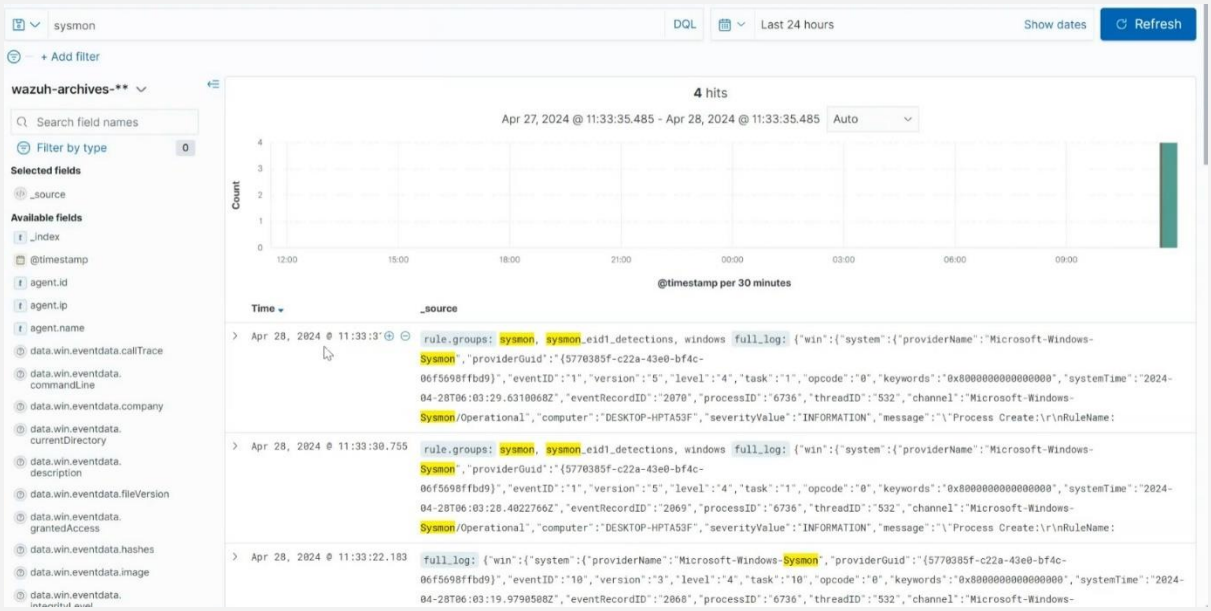


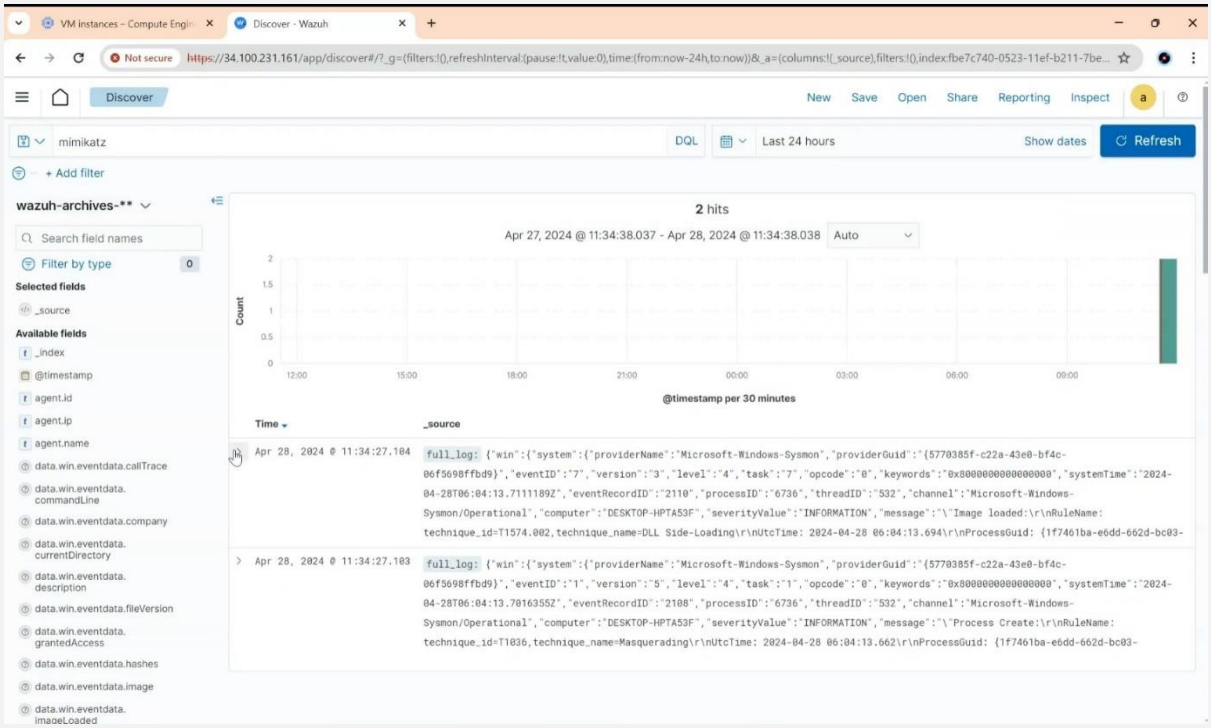
- Now, go to services and restart the wazuh service.

  - Step 5:
- Now, go to SSH of wazuhmanager and find any logs related to mimikatz if detected :
- cat archives.json | grep -i mimikatz



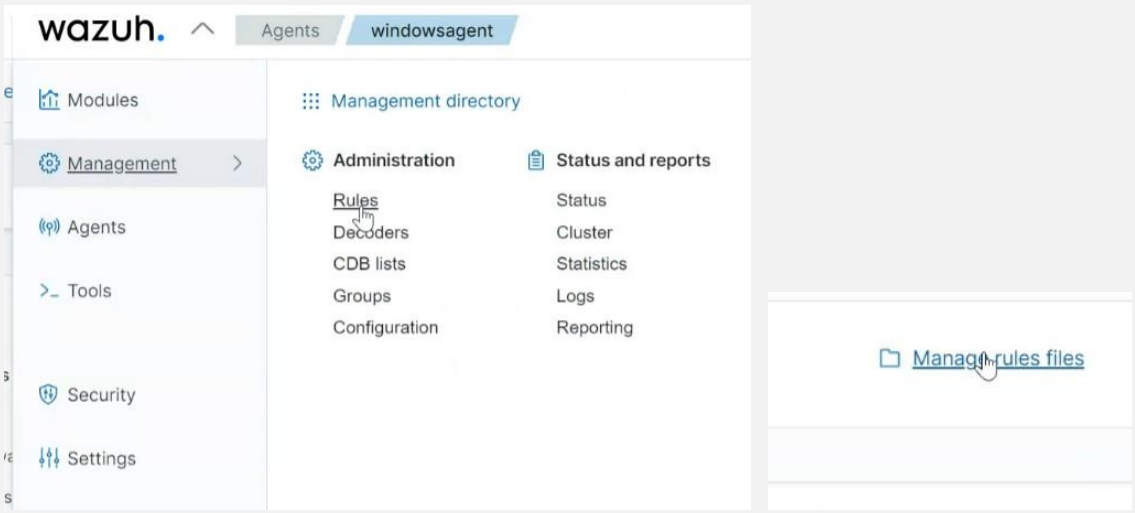
  - Step 6:
- Go to Wazuh Dashboard and at their search for the sysmon tool and mimikatz tool so, at there you'll find the logs related to that event.

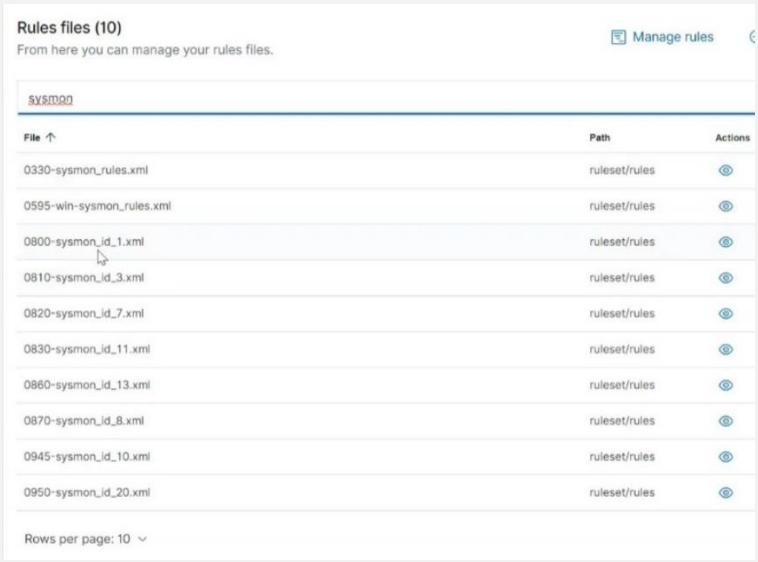


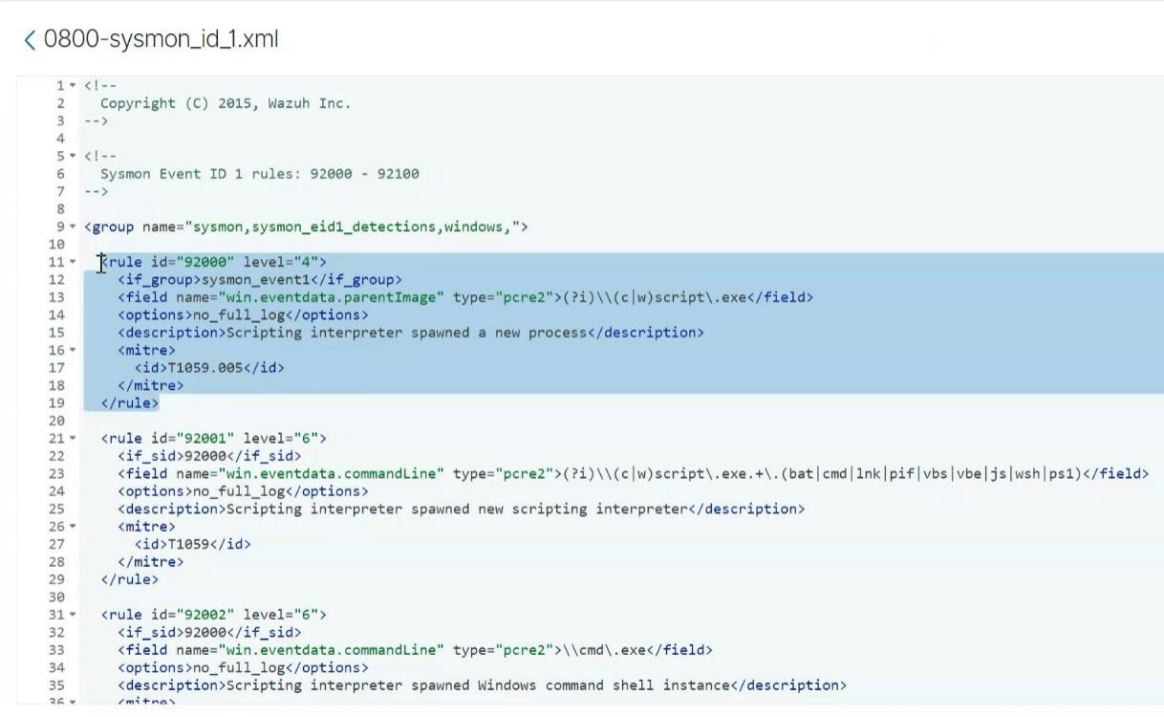


- Step 7:

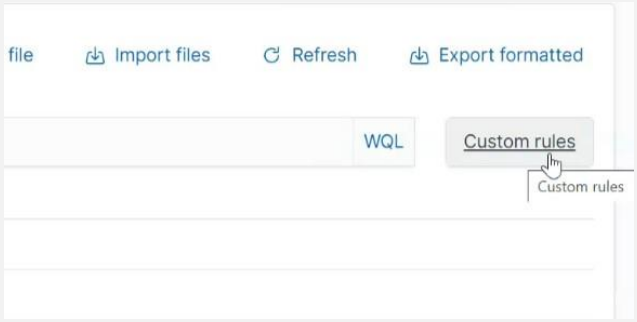
- Now, we'll modify our mimikatz file still want to detect our event because you never know attacker can change the file name and do the attack so for that we'll edit our rules from the Wazuh Dashboard.
- Go to dashboard -> Management -> Administration -> Rules -> Manage rule files -> search for sysmon rule



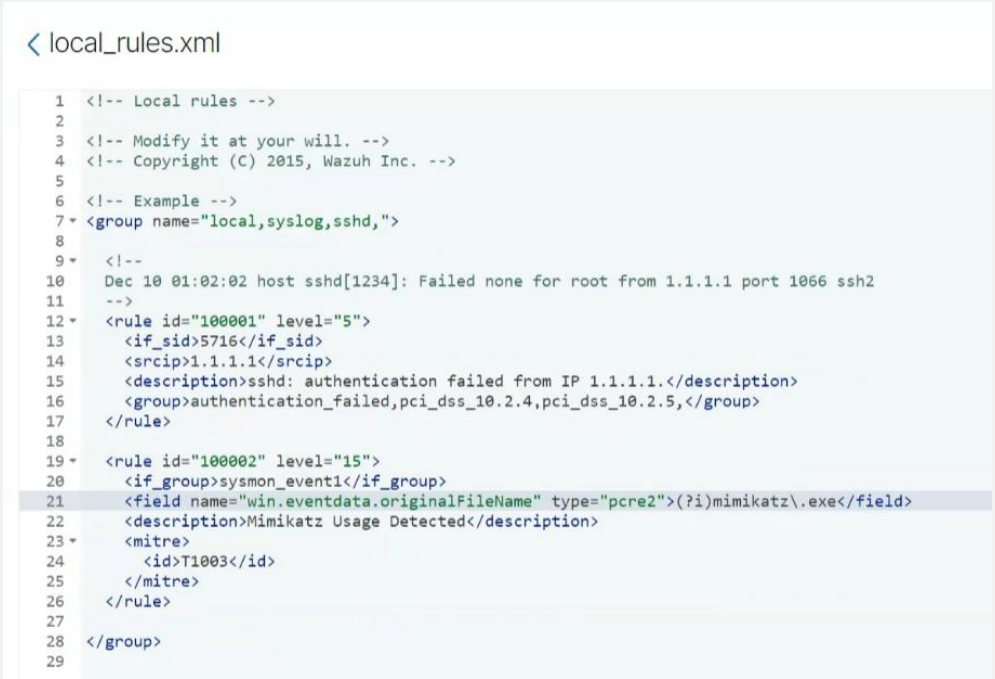




- It will show you a list of sysmon rule; go in any file and copy any rule; Now, go to custom rules -> Edit the local_rules.xml file -> Paste that copied rule here

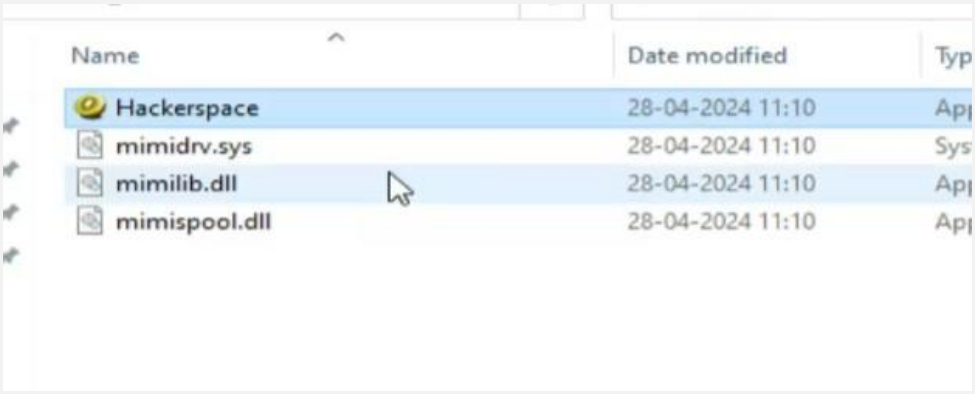


- Now, modified the rule as per your need.



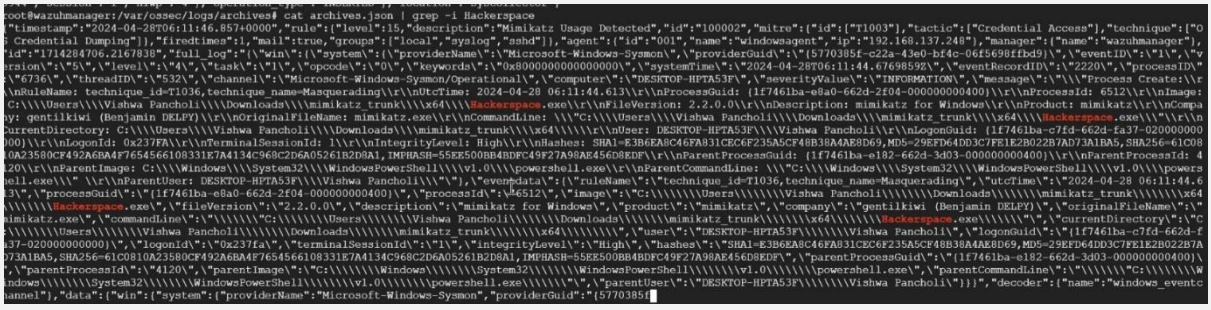
- Save the editing and restart the Manager.

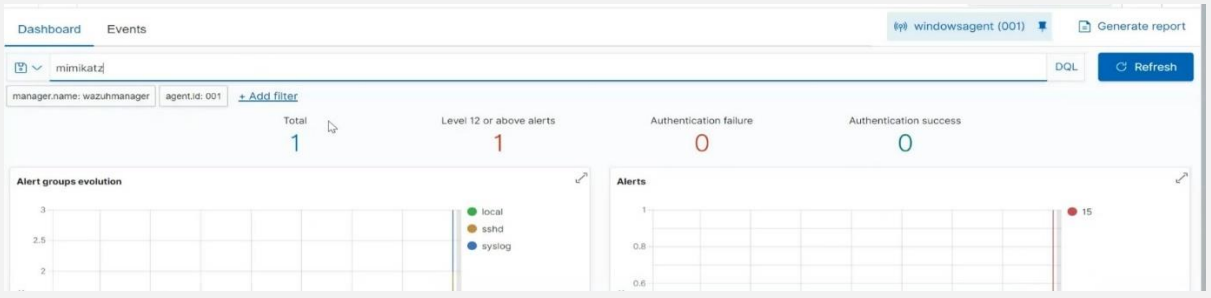
  - Step 8:
- Now, go to the Windows Agent and rename the mimikatz file here I named it as a Hackerspace.



- Now, go to PowerShell and run the hackerspace file.

  - Step 9:
- Now, go to the SSH Manager and run the cat archives.json | grep -i Hackerspace there will be a log of it and go to the Dashboard and search for mimikatz it will show 1 alert named Mimikatz Usage Detected and check originalFilename under logs it will be mimikatz.exe

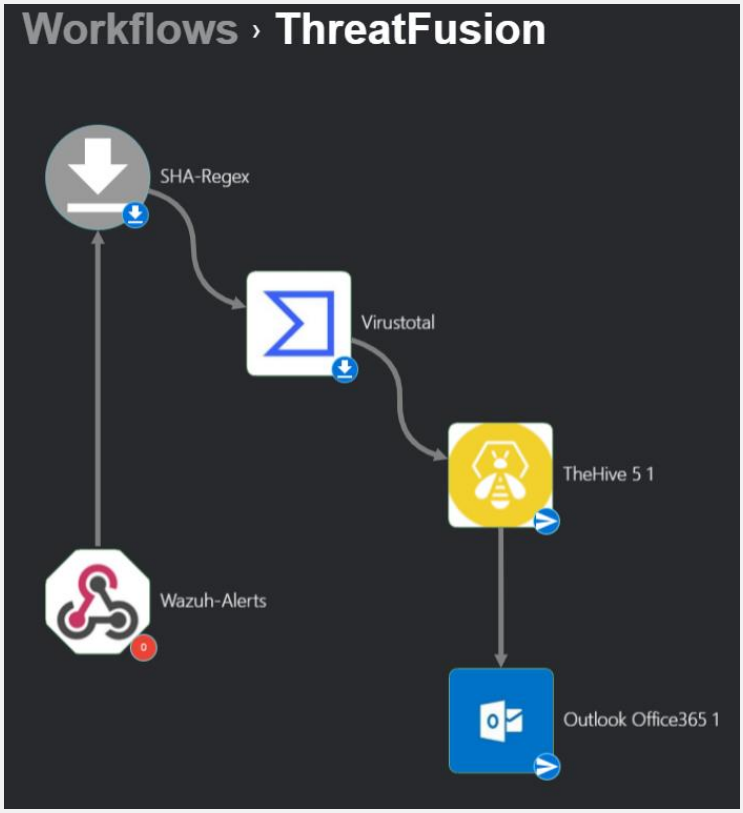




- It means we successfully generated an Alert in our dashboard.

6. Workflow Creation

- In the "Workflow Creation" section, I'll design a workflow utilizing the Shuffle tool, integrating Wazuh and TheHive. Within this workflow, I'll incorporate Virustotal to check the reputation score of detected events. Upon detection of Mimikatz usage by Wazuh, an alert will be dispatched to TheHive, facilitating seamless incident management. The SOC Analyst will then monitor the alert within TheHive, enabling timely investigation and response to security threats. Through this process, the SOC Analyst can efficiently leverage Wazuh, TheHive, and Virustotal to enhance threat detection and incident response capabilities.



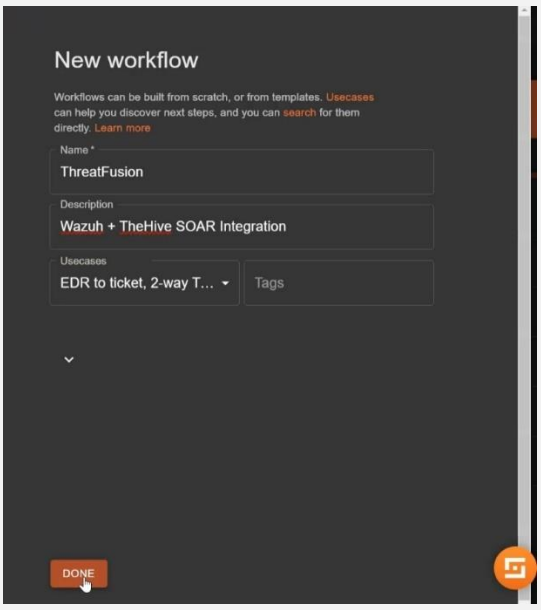
Workflow:

1) Mimikatz Alert Sent to Shuffle
2) Shuffle receives Mimikatz Alert(Extract SHA256 Hash from file)
3) Check Reputation Score with Virustotal
4) Send Details to TheHive to Create Alert
5) Send Email to SOC Analyst to Begin Investigation

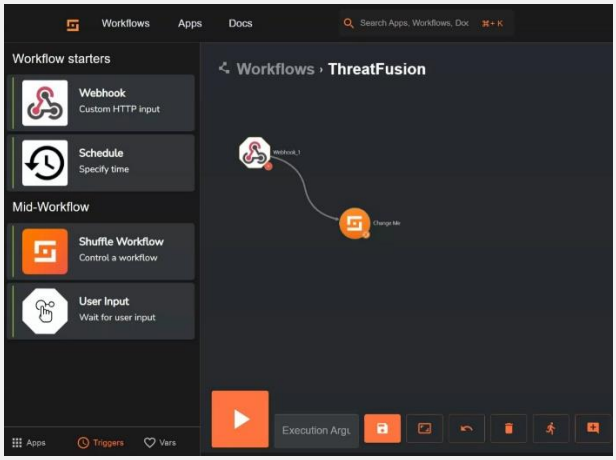
6.1 Handling Mimikatz Alerts: Shuffling Process

- To initiate the process of handling Mimikatz alerts through the Shuffle platform, we'll first create a Shuffle account. This involves signing up on the Shuffle platform's website and configuring account settings. Once the account is set up, we'll proceed to configure Wazuh to redirect alerts to Shuffle. This entails updating the Wazuh manager's configuration to include the Shuffle API endpoint and authentication credentials. Once configured, Wazuh alerts will be automatically forwarded to Shuffle, enabling centralized alert management and workflow automation.

  - Step 1:

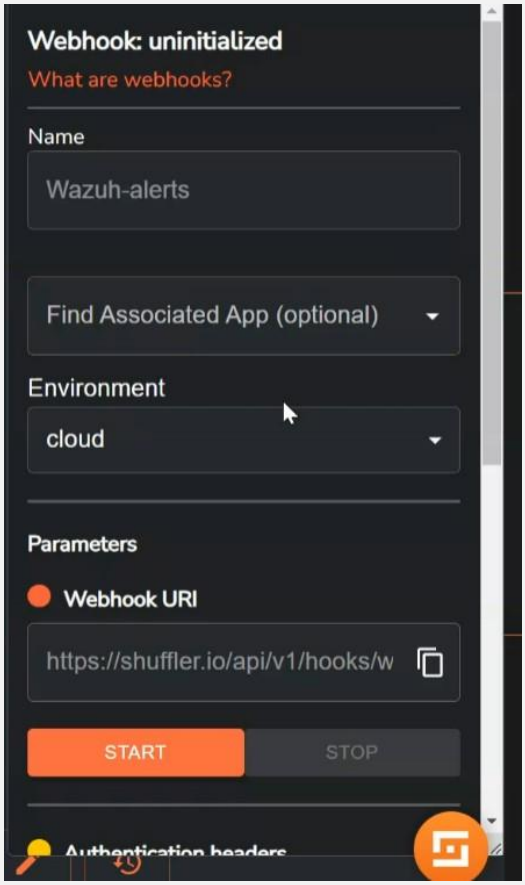


    - We'll Create a workflow named ThreatFusion and add description.



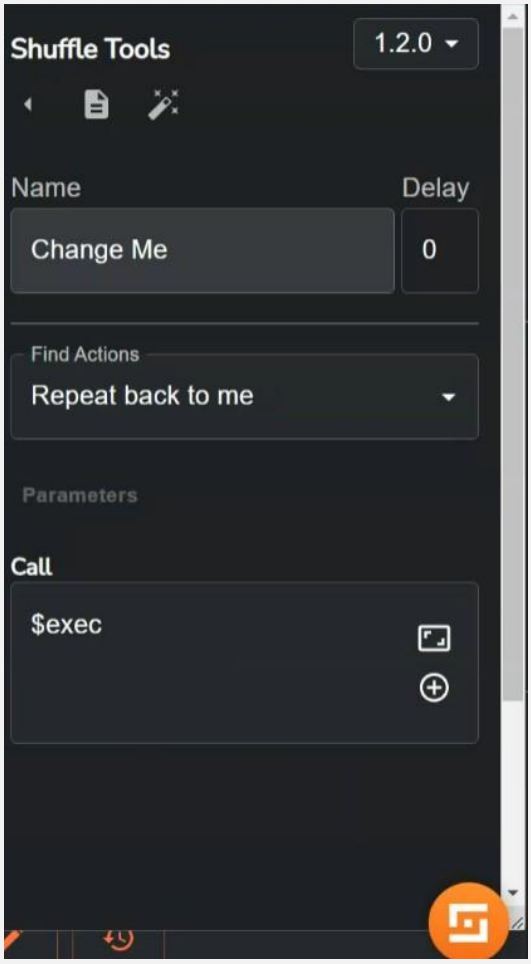
- We'll going to redirect the wazuh alert to shuffle for that you can see there's already change me icon is there we'll add new webhook for wazuh_alerts add that from the triggers

- Step 2:



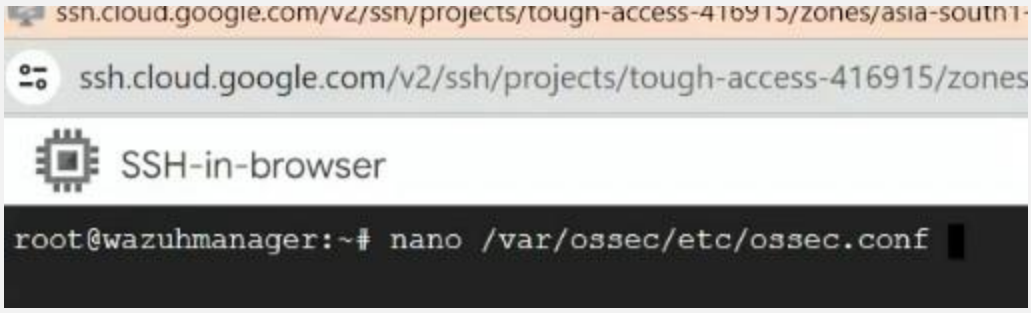
- After adding the webhook we'll modify the name of it and after that save that

- Step 3:



- Now, we are going to modify the change me tool in that add Name, Actions and call which we are going to

- Step 4:
  - Now, go to the wazuhmanager SSH and edit the ossec.conf file for that file path is /var/ossec/etc/ossec.conf

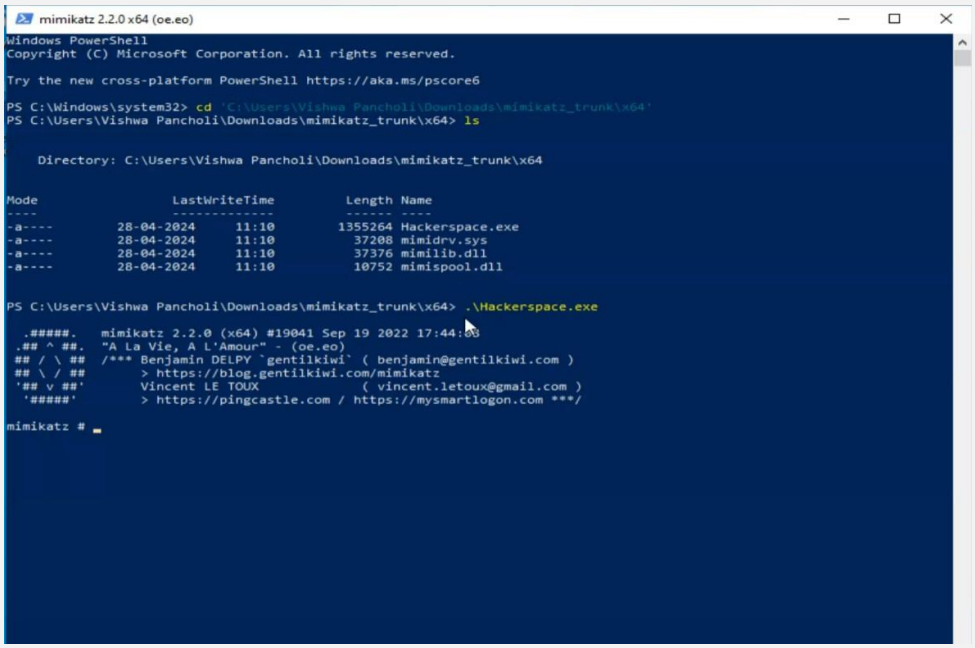


- In that add the below Integration tag in it. Restart the wazuh-manager

```
<integration>
  <name>shuffle</name>
  <hook_url>https://shuffler.io/api/v1/hooks/webhook_2998f026-fae0-4496-b84c-56a2bef914bf </hook_url>
  <rule_id>100002</rule_id>
  <alert_format>json</alert_format>
</integration>
```

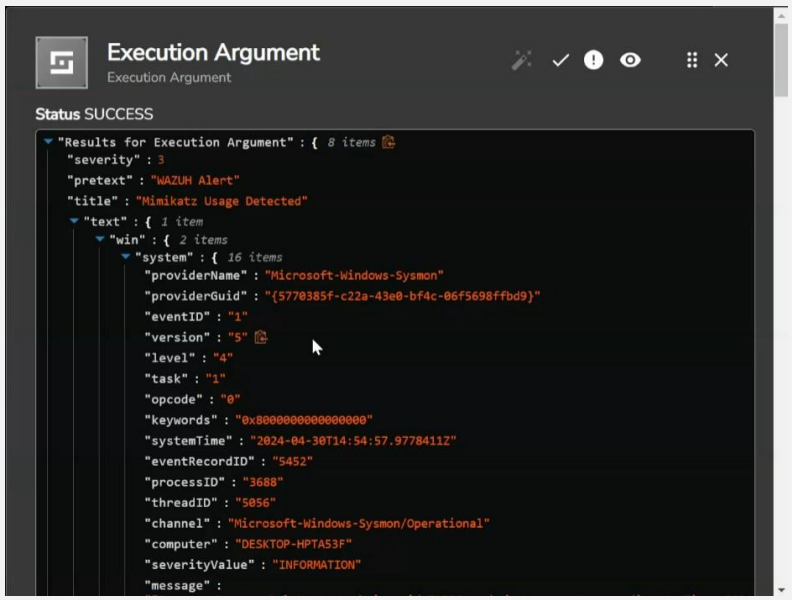
- Step 5:

- Now, go to the agents and run the mimikatz app which we previously named as a Hackerspace



- Step 6:

- Now, save and run the shuffle workflow and then you can see the results.



- In the results you can see there's an Alert which we previously see in wazuh manager

6.2 Converting Alert to Hash

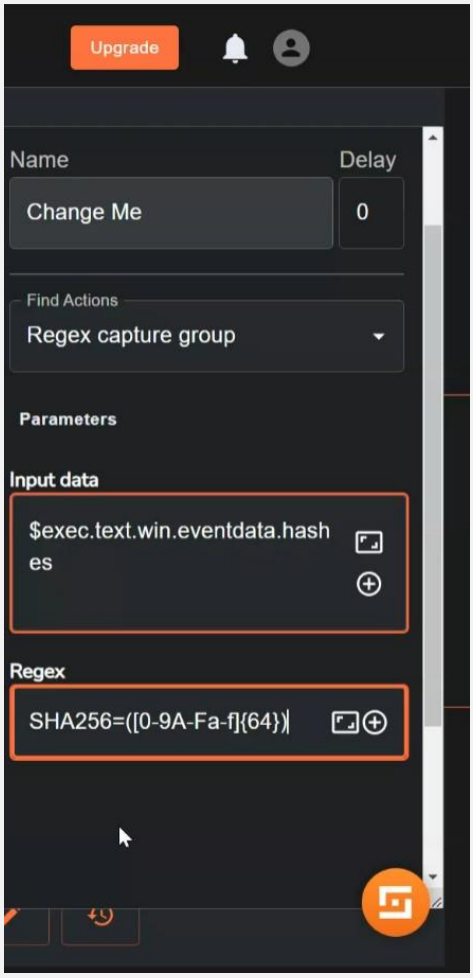
- In the process of converting alerts to a hash format, each incoming alert will undergo transformation into SHA256 hash. This conversion ensures a standardized representation of the alert data, facilitating efficient analysis and comparison. By utilizing the SHA256 hashing algorithm, we create a unique cryptographic hash value for each alert, enhancing data integrity and security within our workflow.

- For the conversion follow the below steps:

  - Step 1:

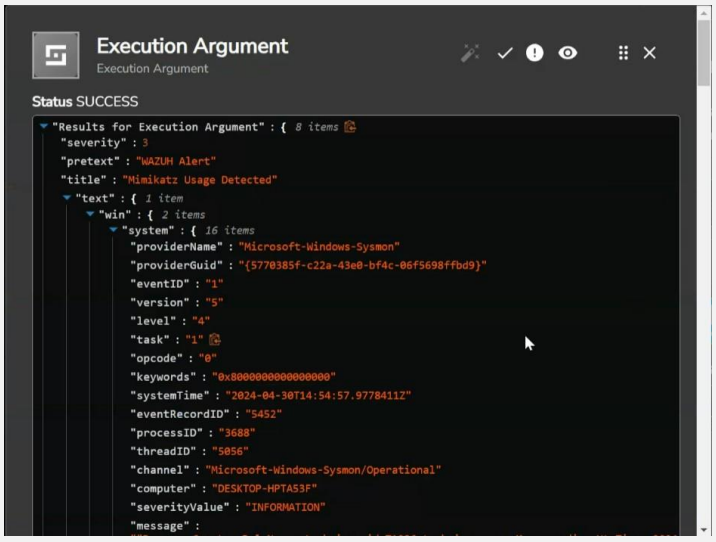
- Now, we'll change the name of Change me as a SHA_Regex and Action will be Regex capture group.

- Now, Regex is a sequence of characters that define a search pattern.

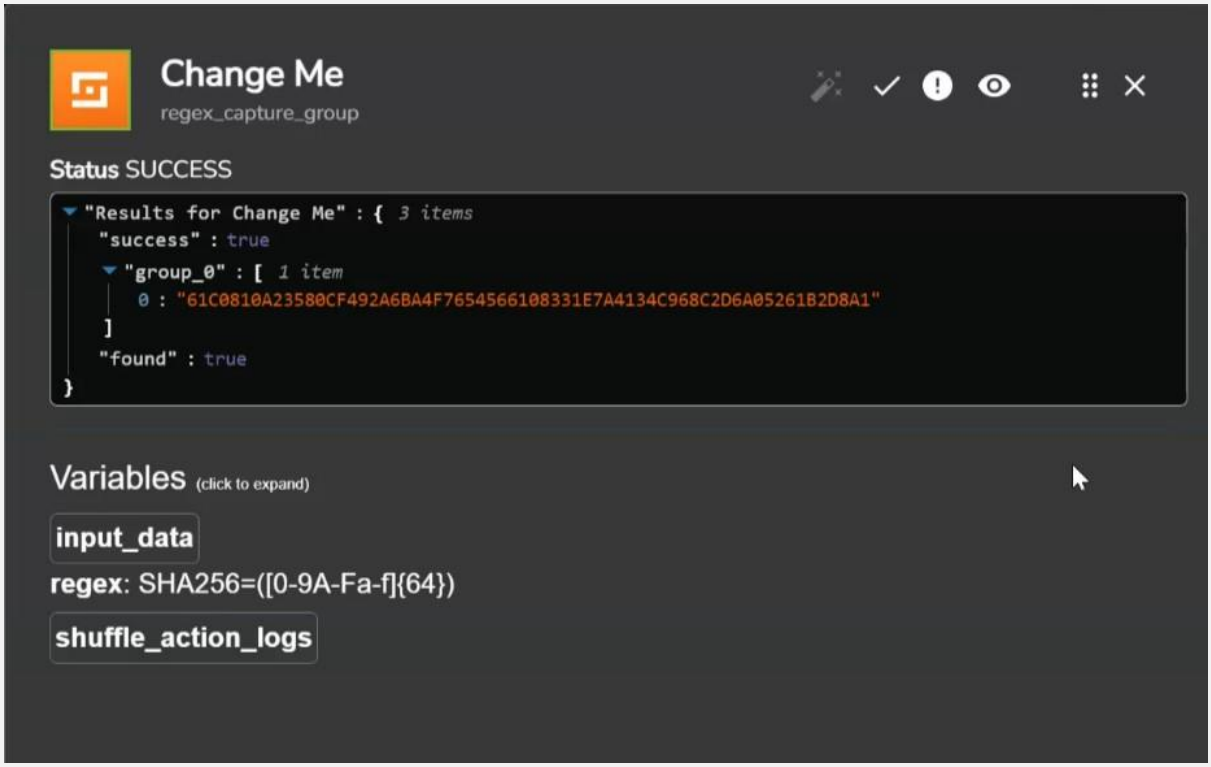


- In the input section we'll add $execution argument -> select hash
- And here regex this value I got from the ChatGPT:

- Step 2:
  - Now, save the workflow and run the execution you'll see the result your alert converted into the hash form.



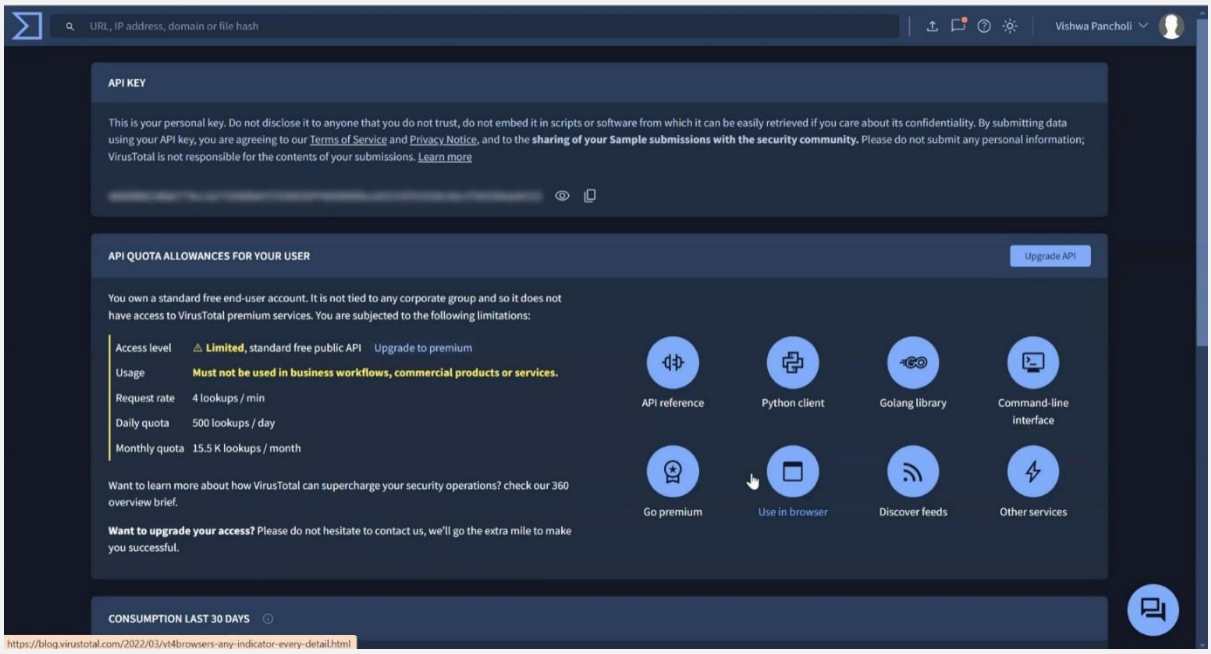
- Step 3:
  - Here you can see the hash formed body of your alert.



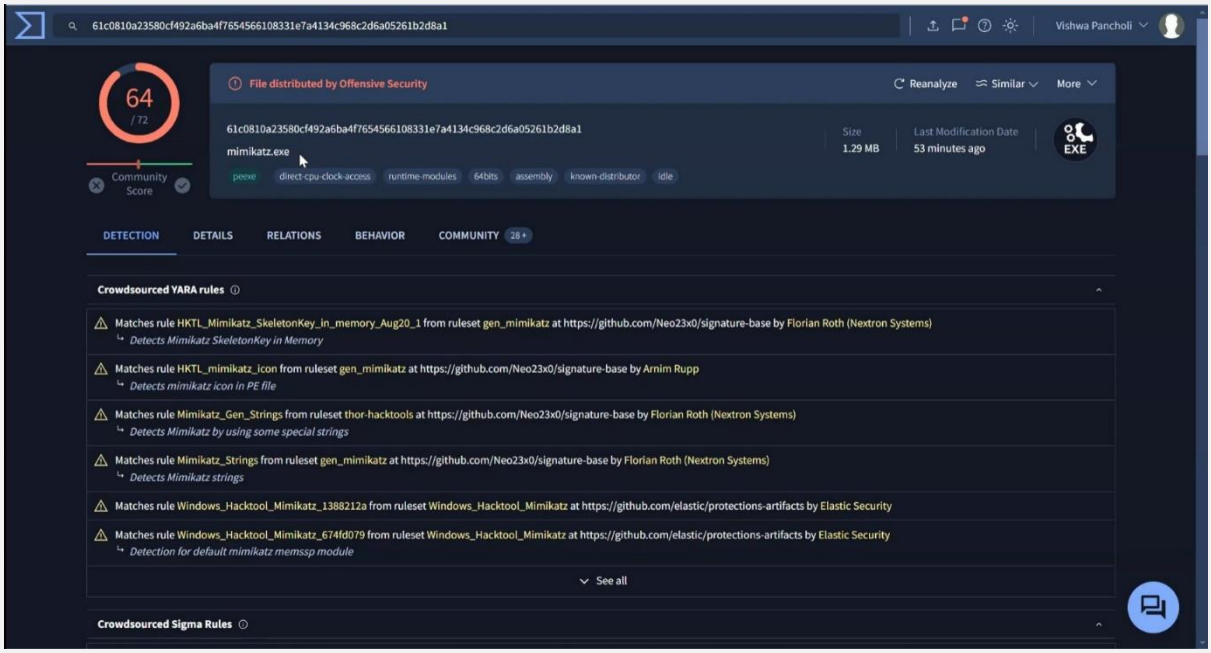
6.3 Verifying Reputation Score via Virustotal

- After converting the alert into a hash format, we will proceed to verify its reputation score using Virustotal. Virustotal is a comprehensive online service that aggregates data from various antivirus engines, website scanners, and file analysis tools to assess the reputation of files and URLs. It provides insights into the trustworthiness and potential threats associated with a given hash or URL. By leveraging Virustotal's extensive database and analysis capabilities, we can augment our threat intelligence efforts and make informed decisions regarding the severity of detected alerts.

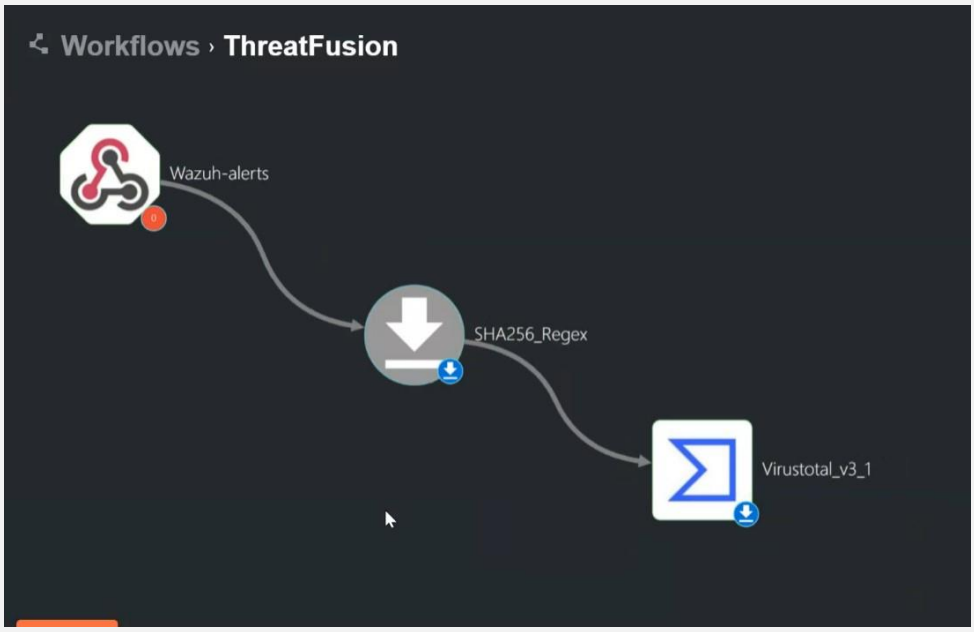
- Step 1:
  - Firstly, create an account on Virustotal and copy the API key.



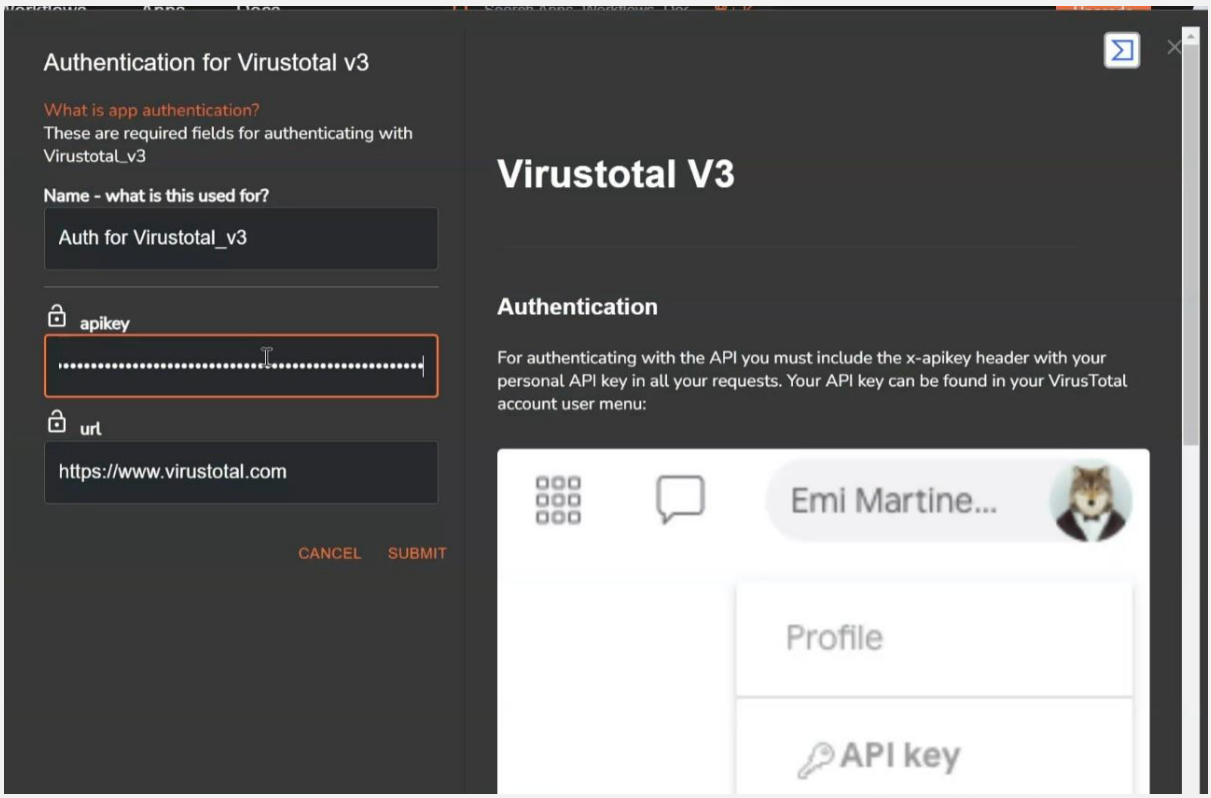
  - Now, in the previous flow we get the hash value copy paste that hash value in the Virustotal and at there you can check the reputation score of that alert.



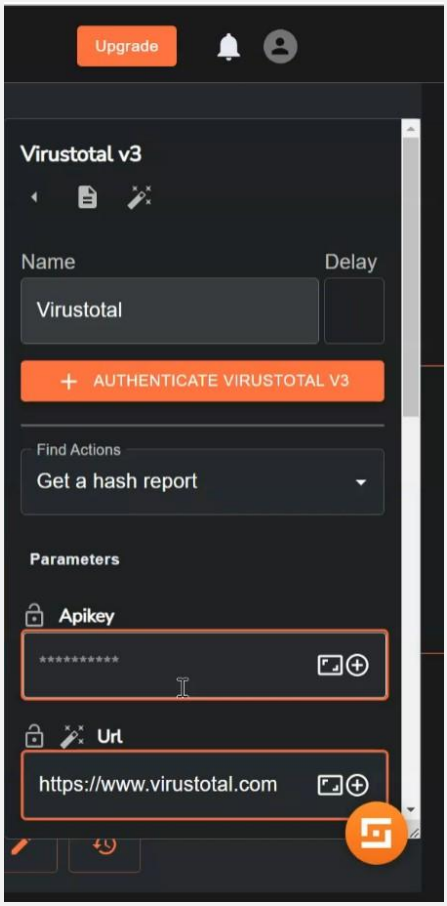
- Step 2:
  - Now, in the shuffle activate the Virustotal app and add it in the workflow.

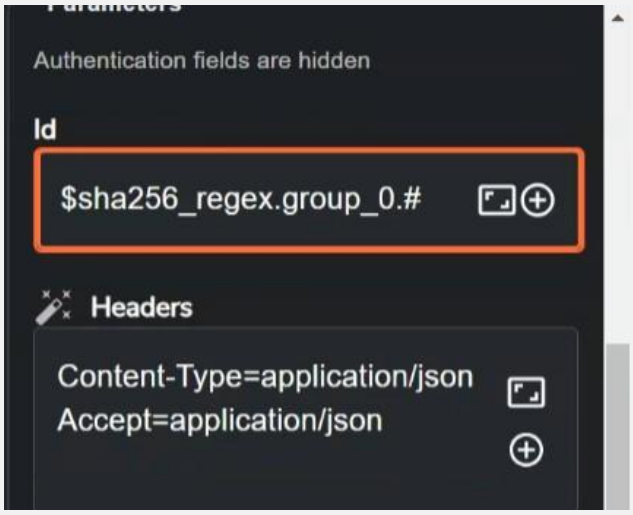


  - Now, Authenticate the Virustotal App using the API key which you got in Virustotal app.

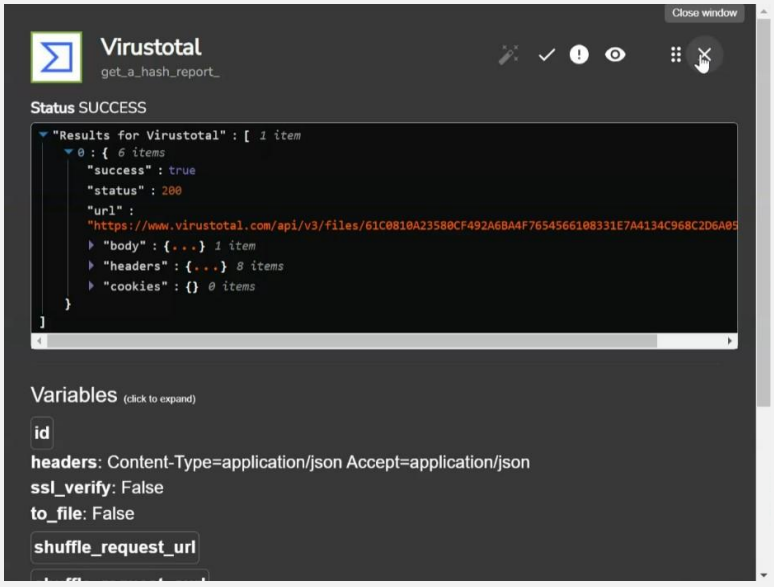


- Step 3:
  - Now, in the ID section you'll add the execution sha_regex.

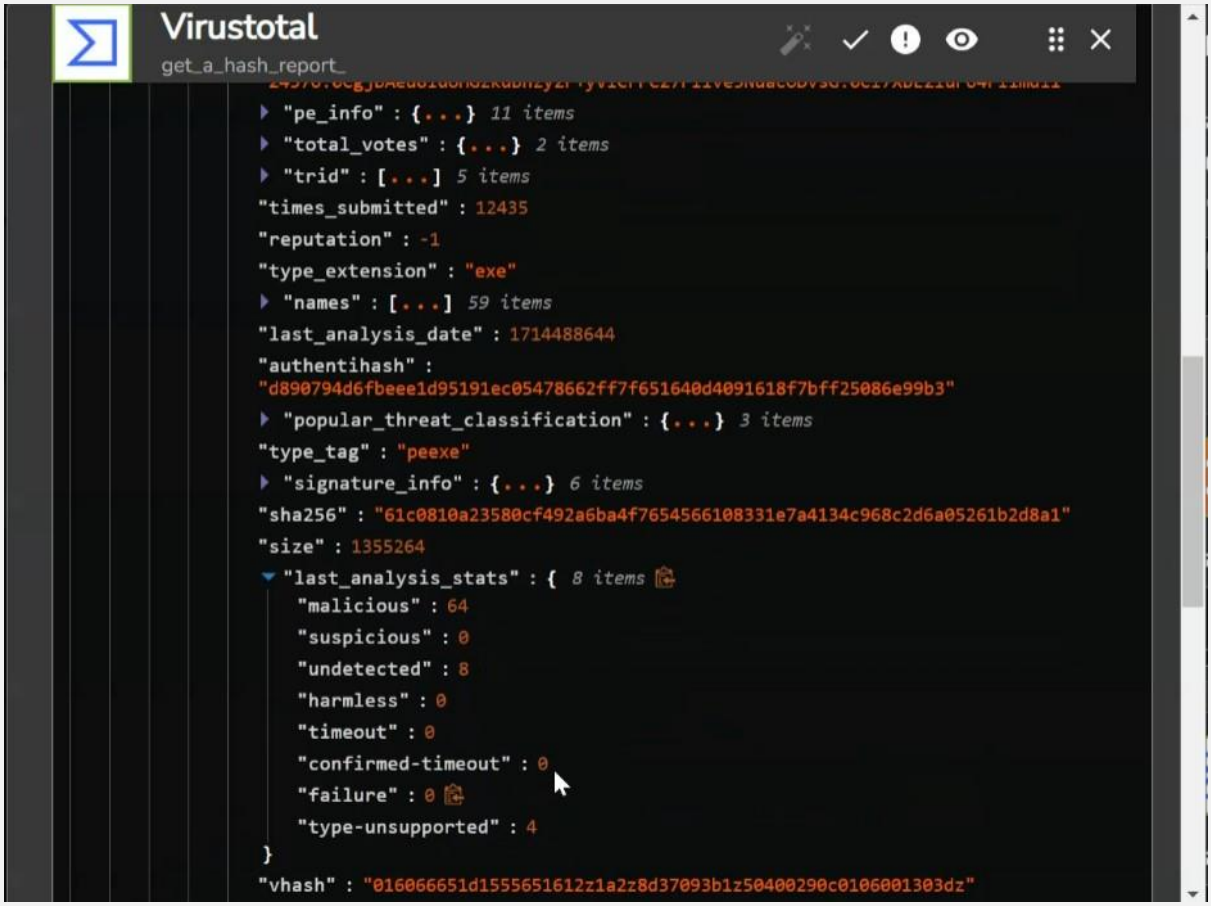




- Step 4:
  - Now, save the workflow and run execution and here you got the URL which redirect you to the Virustotal.



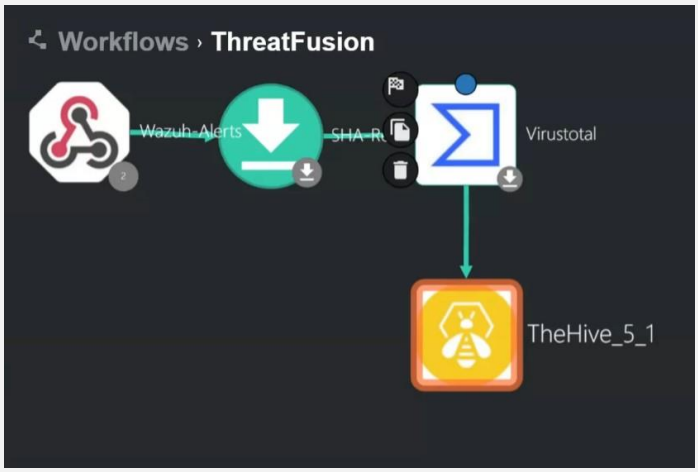
- And also, in the execution result you'll also see the malicious rate under body -> last_analysis_stats



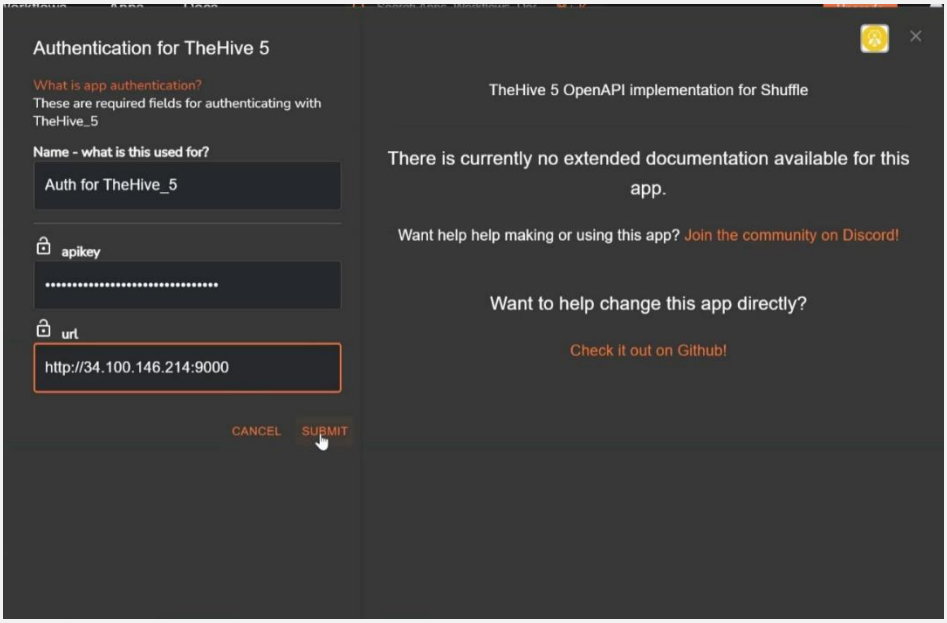
6.4 Creating Alerts in TheHive

- Upon successful redirection of Wazuh alerts, they will be created as alerts within TheHive user interface. TheHive serves as a centralized platform for incident management and response, allowing security analysts to view, triage, and investigate alerts efficiently. By integrating Wazuh with TheHive, we streamline the process of alert handling, enabling seamless collaboration and workflow automation for effective incident response.

- Step 1:
  - Add the TheHive app in the workflow.

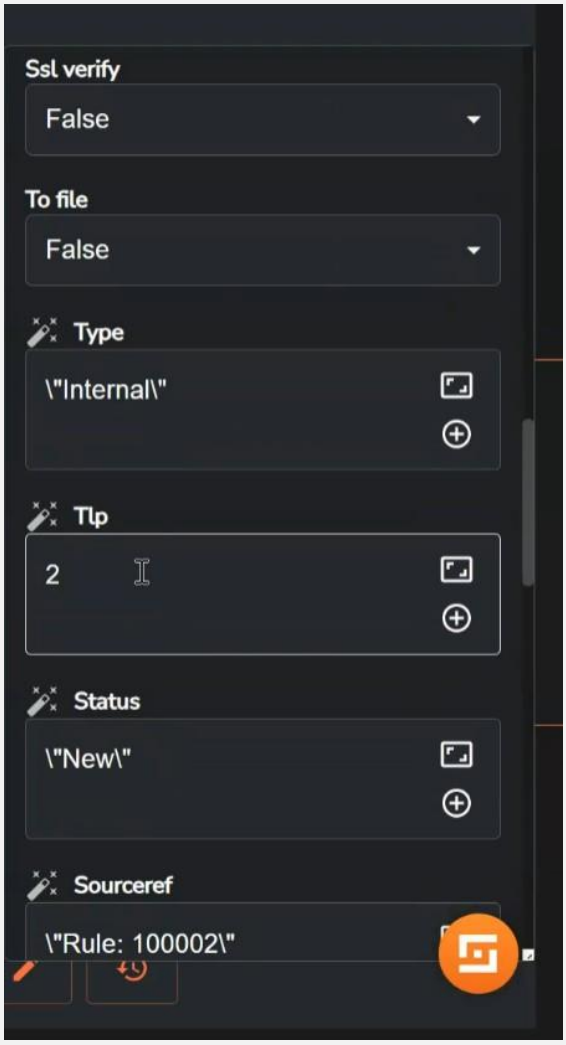


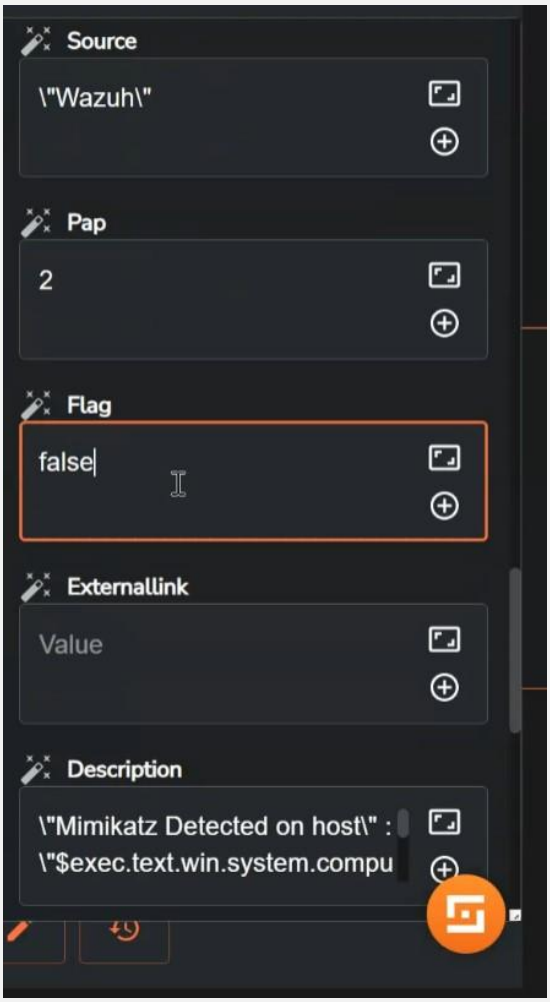
  - Now, authenticate the app using SOAR user API and in URL section add TheHive Public IP Address and Port number 9000.



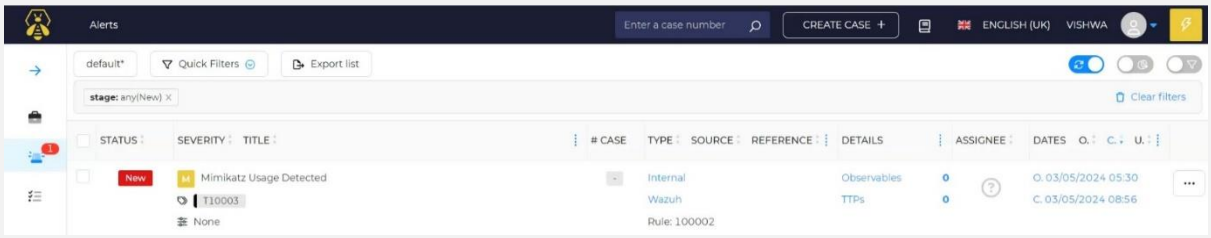
- Step 2:
  - Now, modify the configurations as per you need here I'm adding Tag, Summary, Severity, Title,Summary, Status and description.

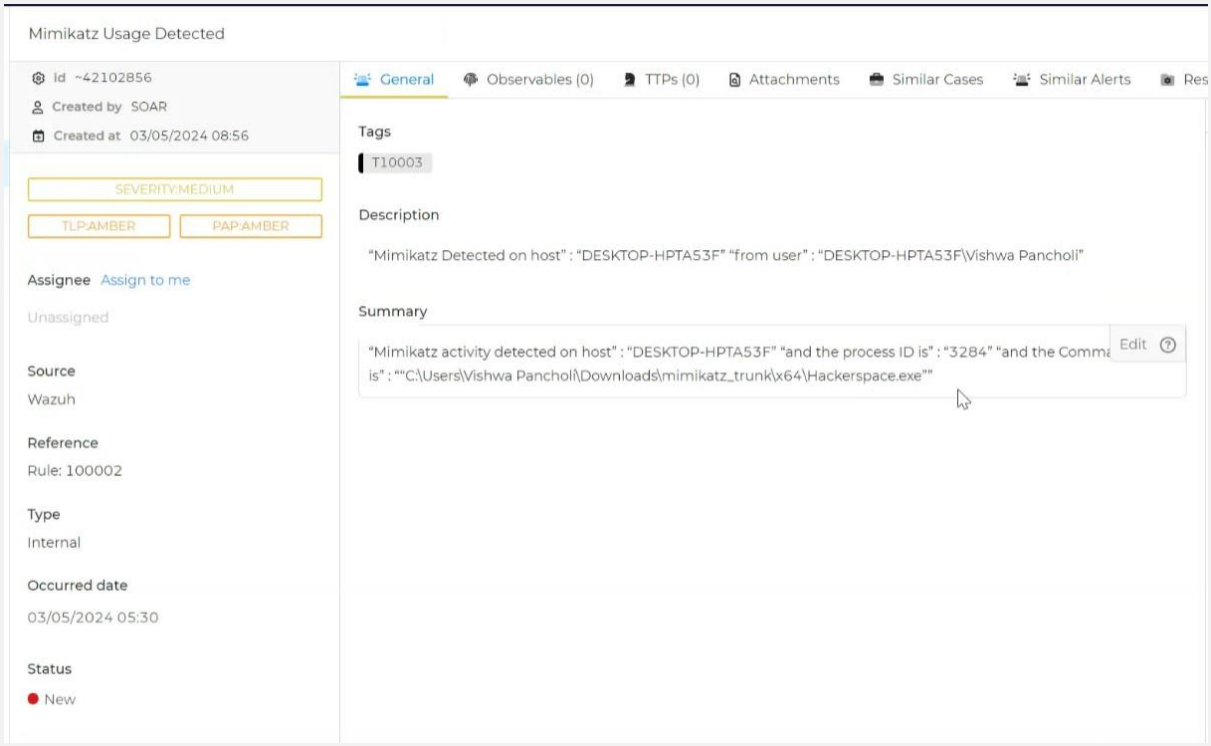

Show Body
Title
\"$exec.title\"
Tags
["T10003"]
Summary
\"Mimikatz activity detected on host\" :
Tags
["T10003"]
Summary
\"Mimikatz activity detected on host\" :
Severity
2
Headers
Content-Type=application/json
Accept=application/json





- Step 3:
  - Here, you can see the alert in the Vishwa User.

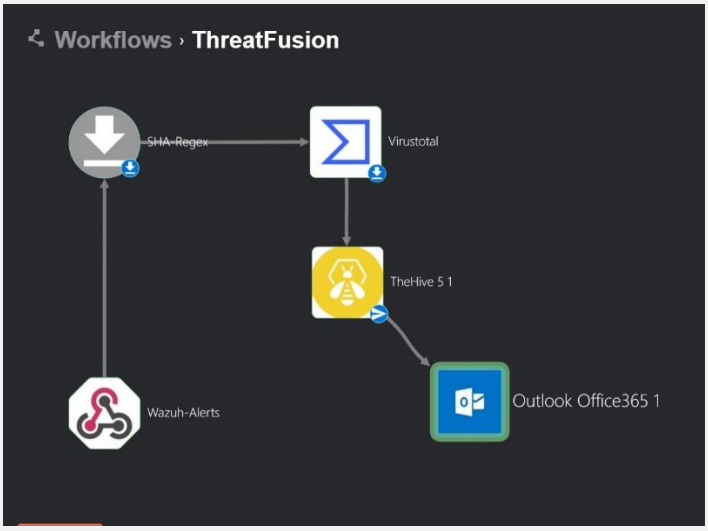




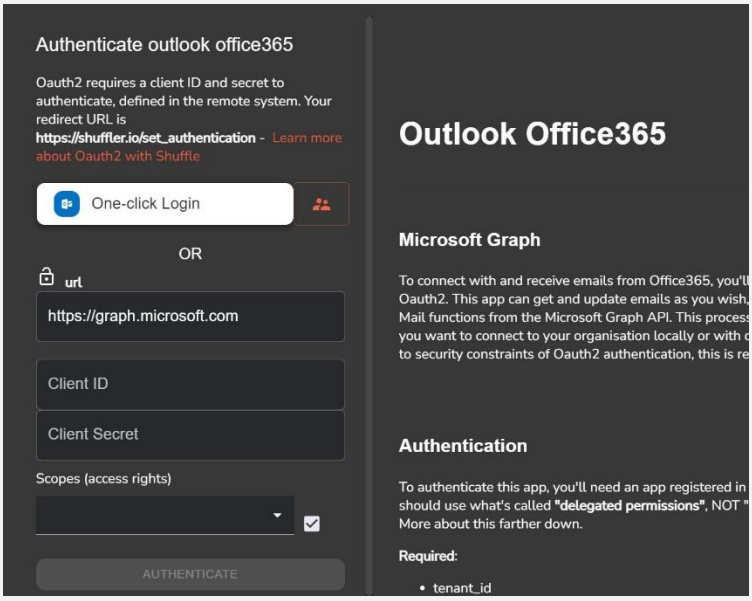
6.5 Email Notification for SOC Analysts to Initiate Investigation

- Upon detection of an alert within TheHive, an email notification will be automatically generated and sent to the SOC Analysts, providing them with pertinent information regarding the alert. This email notification serves as a prompt for SOC Analysts to initiate investigation and response activities. By receiving alerts in their email inbox, SOC Analysts can promptly monitor and assess the alert's severity, enabling timely and effective incident response.

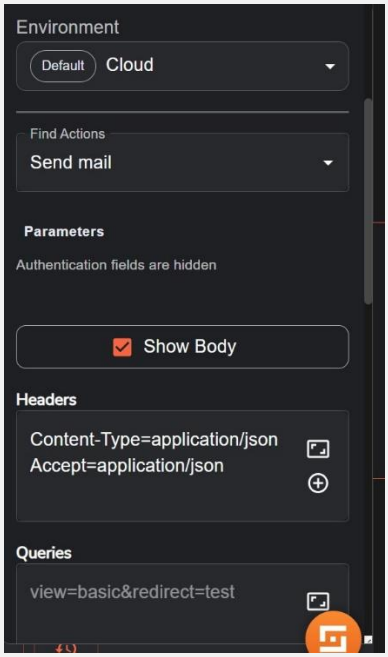
  - Step 1:
    - Add the Outlook office 365 App in the workflow

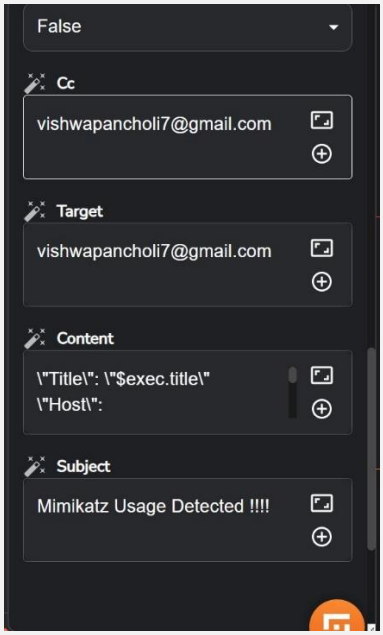


- Step 2:
  - Now, authenticate using One-Click Login and enter your email and code it will authenticate automatically.

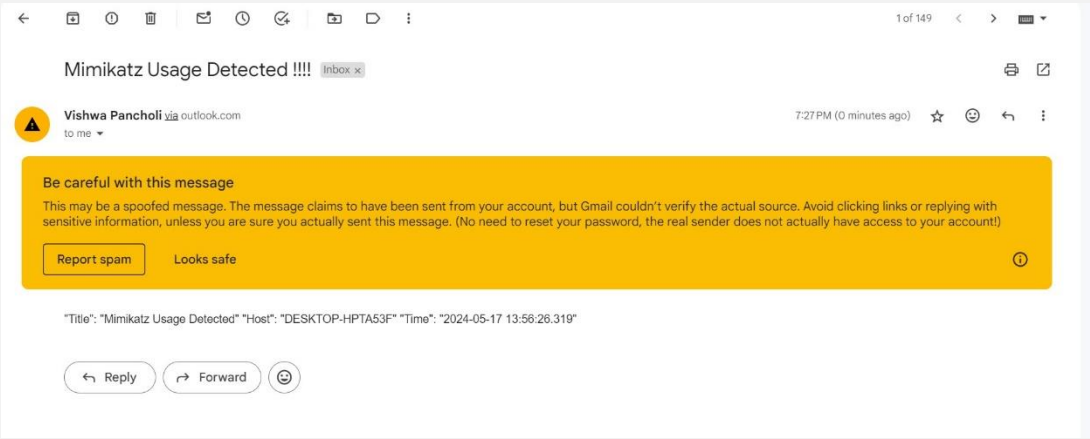


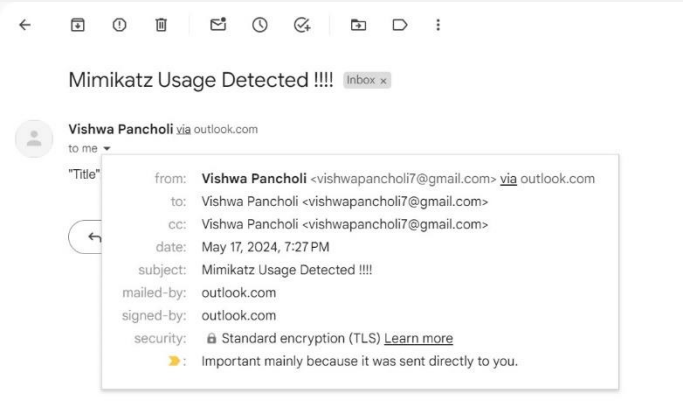
- Step 3:
  - Now, in the action add send email option. Add Subject and Content; and also add the email of SOC Analyst in recipient and cc.





- Step 4:
  - Now, Save the workflow and run the execution mean while go to your windows agent add rerun the hackerspace.exe file and You'll get the email in your inbox named Mimikatz Usage Detected !!!





Conclusion

In conclusion, the integration of Wazuh and TheHive, combined with workflow automation using tools like Shuffle and Virustotal, enhances our ability to detect, triage, and respond to security threats effectively. By centralizing alert management, streamlining incident response workflows, and providing timely notifications to SOC Analysts, we empower organizations to bolster their cybersecurity posture and mitigate risks proactively. This project demonstrates the importance of leveraging comprehensive security solutions and automation tools to safeguard digital assets and ensure the resilience of modern IT environments.

REFERENCES

- Wazuh:
  - https://github.com/MyDFIR/SOC-Automation-Project/blob/main/Wazuh-Install-Instructions

- TheHive:
  - https://github.com/ls111-cybersec/thehive-cortex-misp-docker-compose-lab11update/blob/main/docker-compose.yml

- Sysmon:
  - https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon
    https://github.com/olafhartong/sysmon-modular/blob/master/sysmonconfig.xml

- Mimikatz:
  - https://github.com/gentilkiwi/mimikatz/releases

- https://www.youtube.com/@MyDFIR