Configuration

Basic usage

Command monitoring is configured in the localfile section of ossec.conf. It can be also be centrally configured in agent.conf.

Monitor running Windows processes

Imagine you want to monitor the running process and alert if an important one is not running.

Example with notepad.exe:

1. Configure the agent to accept remote commands from the manager, in the agent's **local_internal_options.conf**.

```
# Logcollector - Whether or not to accept remote commands from the manager
logcollector.remote_commands=1
```

2. Define the command to list running processes, in the manager's **agent.conf** file

```
<localfile>
     <log_format>full_command</log_format>
     <command>tasklist</command>
     <frequency>120</frequency>
 </localfile>
```

The **frequency** defines how often the command will be run (in seconds).

3. Define the rules

```
<rule id="100010" level="6">
  <if_sid>530</if_sid>
  <match>^ossec: output: 'tasklist'</match>
  <description>Important process not running.</description>
  <group>process_monitor,</group>
</rule>
<rule id="100011" level="0">
  <if_sid>100010</if_sid>
  <match>notepad.exe</match>
  <description>Processes running as expected</description>
  <group>process_monitor,</group>
</rule>
```

The first rule (100010) will generate an alert ("Important process not running"), unless it is overridden by its child rule (100011) that matches *notepad.exe* in the command output. You could add as many additional child rules as you need to enumerate all important processes you want to monitor. You could also adapt this example to monitor Linux process by changing the <command> from "tasklist" to a Linux command that lists processes, like "ps -auxw".

Disk space utilization

We can configure the `dh` command in the manager's **agent.conf** file or in the agent's **ossec.conf** file:

```
<localfile>
    <log_format>command</log_format>
    <command>df -P</command>
</localfile>
```

Wazuh already has a rule to monitor this:

```
<rule id="531" level="7" ignore="7200">
  <if_sid>530</if_sid>
  <match>ossec: output: 'df -P': /dev/</match>
  <regex>100%</regex>
  <description>Partition usage reached 100% (disk space monitor).</description>
  <group>low_diskspace,pci_dss_10.6.1,</group>
</rule>
```

The system will alert once the disk space usage on any partition reaches 100%

Check if the output changed

In this case we use the Linux “netstat” command and the check_diff option to monitor for changes in listening tcp sockets.

Configuration in `agent.conf` or `ossec.conf`:

```
<localfile>
  <log_format>full_command</log_format>
  <command>netstat -tan |grep LISTEN|grep -v 127.0.0.1</command>
</localfile>
```

Wazuh already has a rule to monitor this:

```
<rule id="533" level="7">
  <if_sid>530</if_sid>
  <match>ossec: output: 'netstat -tan</match>
  <check_diff />
  <description>Listened ports status (netstat) changed (new port opened or closed).</description>
  <group>pci_dss_10.2.7,pci_dss_10.6.1,</group>
</rule>
```

If the output changes, the system will generate an alert, indicating a network listener has disappeared or a new one has appeared, which could indicate something is broken or a network backdoor has been installed.

Load average

You can configure Wazuh to monitor the Linux `uptime` command and alert when it is higher than a given threshold, like 2 in this example.

Configuration in `agent.conf` or `ossec.conf`:

```
<localfile>
    <log_format>command</log_format>
    <command>uptime</command>
</localfile>
```

And the custom rule to alert when is higher than 2:

```
<rule id="100101" level="7" ignore="7200">
  <if_sid>530</if_sid>
  <match>ossec: output: 'uptime': </match>
  <regex>load averages: 2.</regex>
  <description>Load average reached 2..</description>
</rule>
```

Detect USB Storage

Wazuh can be configured to alert when a USB storage device is connected. This example is for a Windows agent.

Configure your agent to monitor the USBSTOR registry entry, by adding this to the manager's `agent.conf` :::

```
<agent_config os="Windows">
  <localfile>
      <log_format>full_command</log_format>
      <command>reg QUERY HKLM\SYSTEM\CurrentControlSet\Enum\USBSTOR</command>
  </localfile>
</agent_config>
```

Next create your custom rule:

```
<rule id="140125" level="7">
    <if_sid>530</if_sid>
    <match>ossec: output: 'reg QUERY</match>
    <check_diff />
    <description>New USB device connected</description>
</rule>
```