Translated from Spanish to English - www.onlinedoctranslator.com

a

Manual for creating a Security Information and Event Manager (SIEM)

Ethical hacking

Engineer Andrés Zambrano - Engineer Kevin Rodríguez - Engineer Sebastián Reina Balcázar

table of Contents

Objective of this Manual **3**

Minimum System Requirements **3**

Recommended System Requirements **3**

Installation **4**

Install on Linux *4*

Install in windows *5*

Downloading, unzipping, and installing the Elastic Agent *6*

Option 1 6

Option 2 7

Validating that machine logs are being received correctly *9*

Uninstall Elastic on virtual machines **11**

Installation Problems **11**

Getting Started with the Product - Generating Events on Kali and Windows **14**

Generating events in Kali and viewing them in the Elastic Log console *14*

Generating events on Windows 10 and displaying them in the Elastic Log console *17*

Installation Manual

Objective of this Manual

This manual is intended to serve as a guide for those interested in working with a Security Information and Event Manager (SIEM) so they can easily install the tools needed to use it.

Minimum System Requirements

System Processor:	*4 CPU*
System Memory:	*8 GB RAM*
Free Disk Space:	*50 GB*
Operating System:	*Windows 10 x64 – Kali Linux*

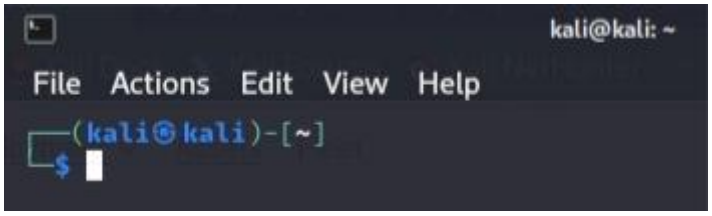
Recommended System Requirements

System Processor:	*6 CPU*
System Memory:	*24 GB RAM*
Free Disk Space:	*200 GB*
Operating System:	*Windows 10 x64 or higher – Kali Linux*

Facility

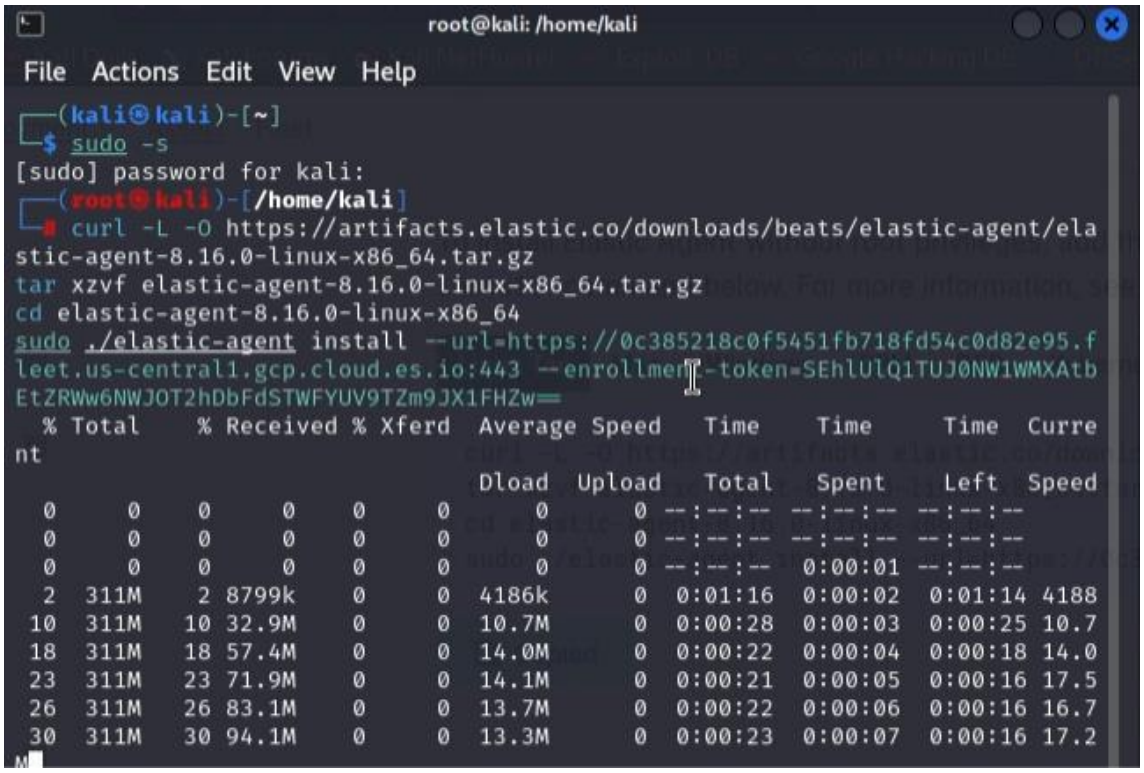
Install on Linux

The first step is to install the agent on the Kali Linux virtual machine. To do this, open a command console..

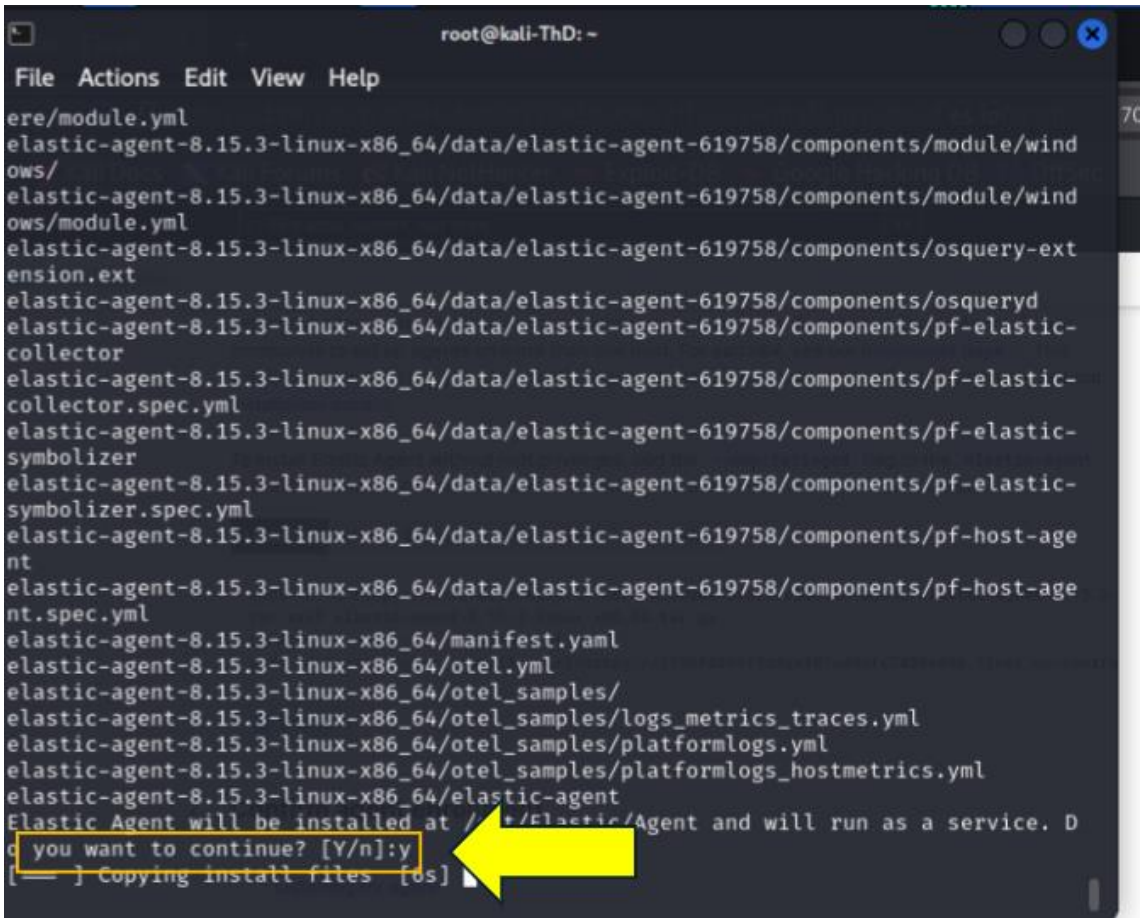


You must make sure you have administrator permissions

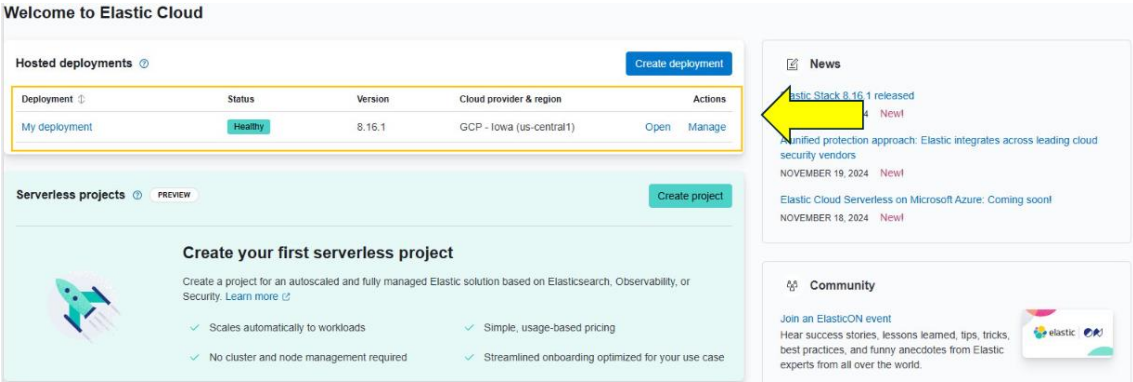
Now we paste the command and the installation will start.



When it finishes installing all the dependencies, we type the “Y” key and then “Enter”.



If we go back to the page of *elastic* we will see that the integration was successful.

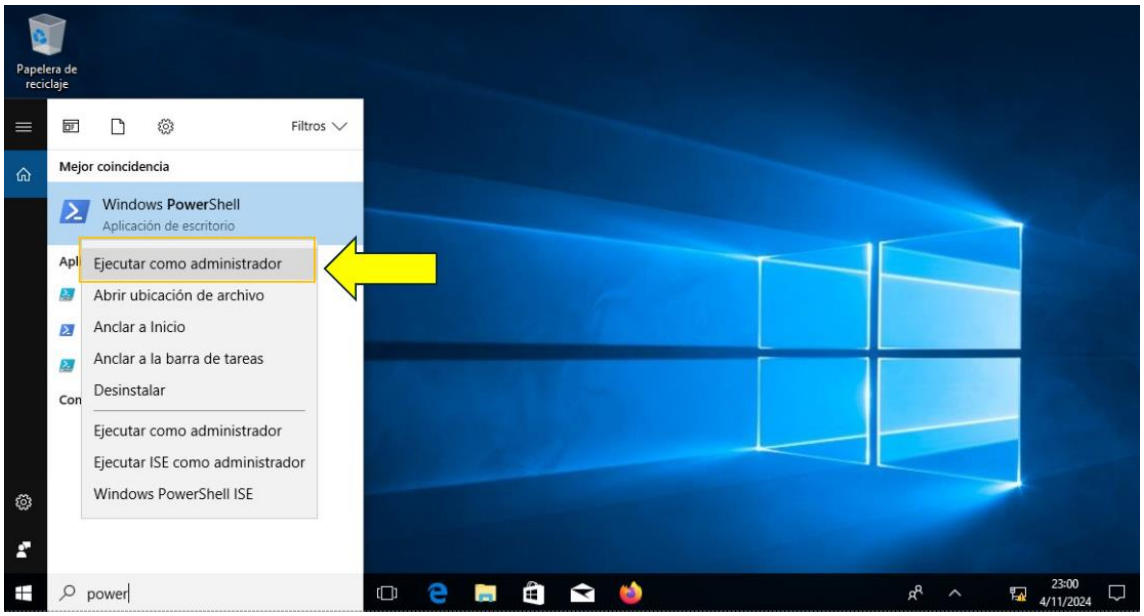


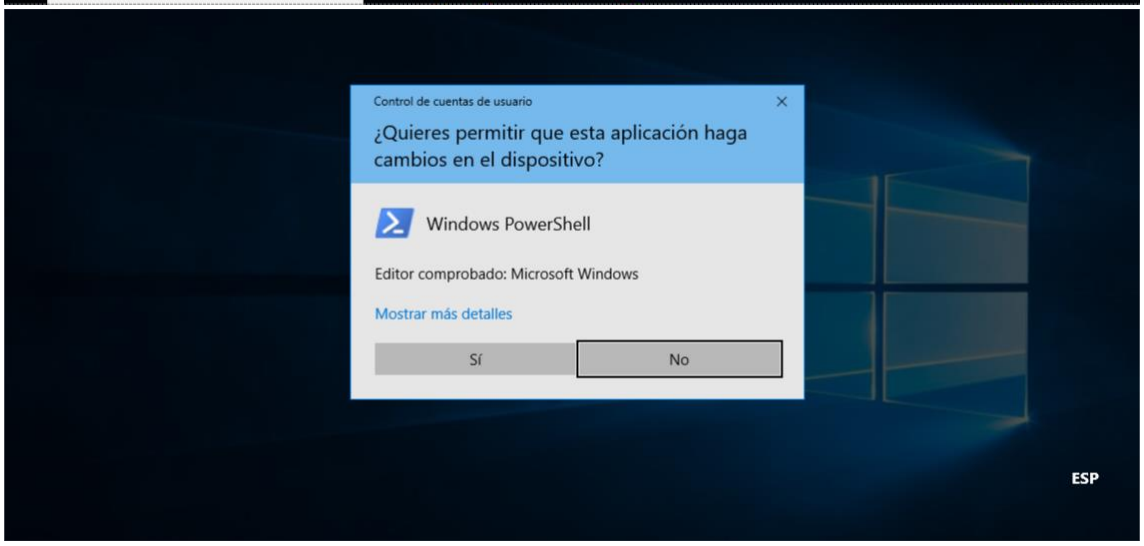
Install on Windows

For Elastic integration on Windows, we will use a Windows 10 x64 virtual machine.

The machine we used in this tutorial runs on AMD because the host uses AMD architecture. This information is important to keep in mind when downloading and installing the Elastic Agent on Windows.

For integration, we can use either the Command Prompt (CMD) or Windows PowerShell. In this case, we'll use PowerShell, which is the option recommended by Windows. When opening it, make sure you have administrator permissions.



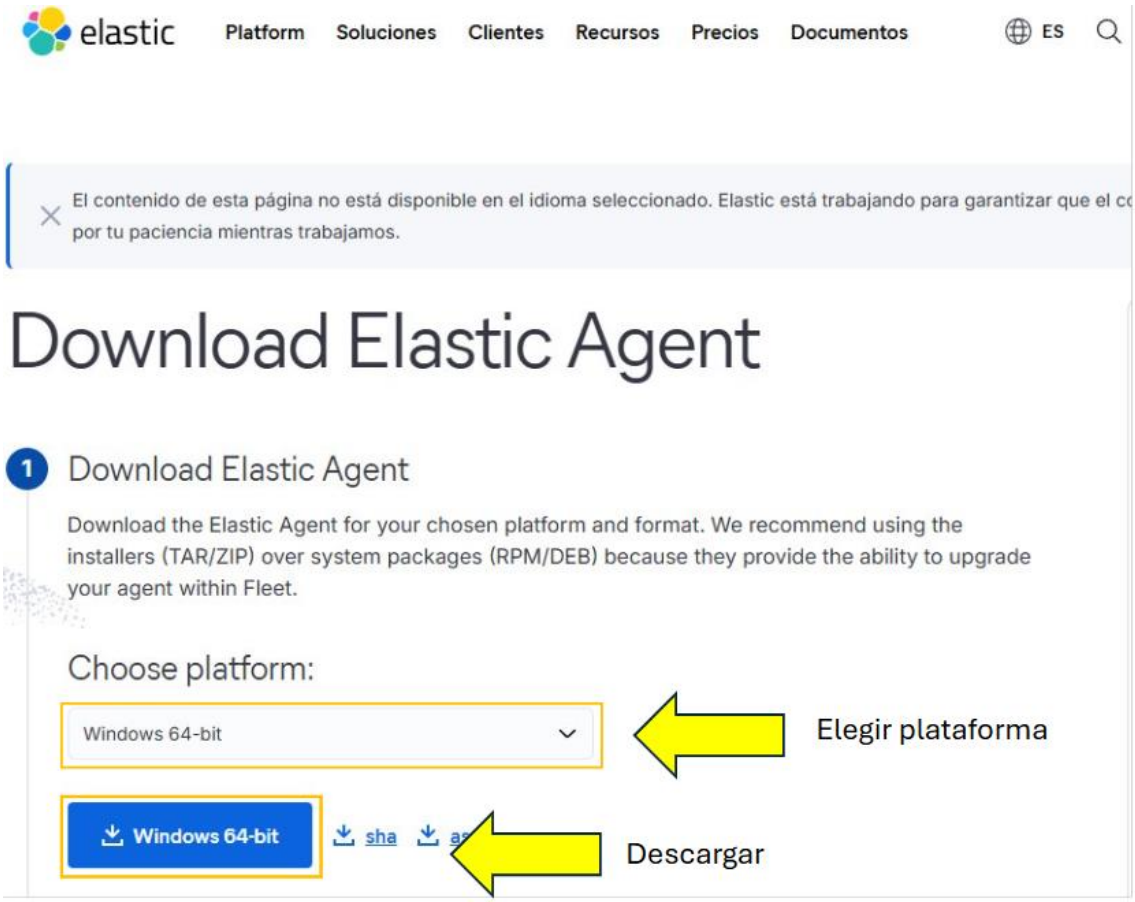


Download, unzip, and install Elastic Agent

Option 1

Download directly from the official elastic.co website. We'll choose the platform we'll be working on; in this case, we'll use a Windows 10 virtual machine with 64-bit architecture.

We can download the "elastic agent" directly from the Elastic website. Type "elastic agent" in the search engine and click "Download." This will automatically download the latest version of the Elastic agent.



Option 2

From the “PowerShell” command line we will write the following command “$ProgressPreference = 'SilentlyContinue' ”

This command downloads the installer for a machine running Windows 10 and an AMD processor. For other cases, follow the instructions in the Elastics installation manual.

Next, you need to copy the following command into the console “Invoke-WebRequest -URI https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-8.15.3-windowsx86_x64.zip -Outfile elastic-agent-8.15.3 windows-x86_x64.zip”

```
PS C:\> Invoke-WebRequest -URI https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-8.15.2-windows-x86_x64.zip -OutFile elastic-agent-8.15.2-wi
ndows-x86_x64.zip
```

This unzips the downloaded .zip file and saves it to the desired folder. In this case, we're using the system32 folder, located in the Windows folder. "Expand-Archive .\elastic-agent-8.15.3-windows-x86_x64.zip -C:\Windows\system32"

```
PS C:\Windows\system32> Expand-Archive .\elastic-agent-8.15.2-windows-x86_x64.zip C:\Windows\system32
```

We go to the folder where we unzipped the .zip file with the installer and verify that the executable file “elastic-agent.exe” exists.

```
PS C:\Windows\system32> cd elastic-agent-8.15.2-windows-x86_x64
PS C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64>
```

```
PS C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64> dir

    Directorio: C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        11/6/2024  11:27 AM                elastic-agent-8.15.3+build202411051926-windows-x86_64

PS C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64> cd elastic-agent-8.15.3+build202411051926-windows-x86_64
PS C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64\elastic-agent-8.15.3+build202411051926-windows-x86_64> dir

    Directorio:
    C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64\elastic-agent-8.15.3+build202411051926-windows-x86_64

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        11/6/2024  11:27 AM                data
-a----        11/5/2024   8:09 PM             41 .build_hash.txt
-a----        11/5/2024   8:09 PM             41 .elastic-agent.active.commit
-a----        11/5/2024   8:09 PM       53515200 elastic-agent.exe
-a----        11/5/2024   8:09 PM          14829 elastic-agent.reference.yml
-a----        11/5/2024   8:09 PM          12306 elastic-agent.yml
-a----        11/5/2024   8:09 PM          13675 LICENSE.txt
-a----        11/5/2024   8:09 PM            393 manifest.yaml
-a----        11/5/2024   8:09 PM        3388637 NOTICE.txt
-a----        11/5/2024   8:09 PM            643 otel.yml
-a----        11/5/2024   8:09 PM             88 otelcol.ps1
-a----        11/5/2024   8:09 PM             25 package.version
-a----        11/5/2024   8:09 PM            387 README.md

PS C:\Windows\system32\elastic-agent-8.15.2-windows-x86_x64\elastic-agent-8.15.3+build202411051926-windows-x86_64>
```

We enter the following command to run the .exe file and start the installation.

In the url and token we will use the same ones that were used in the installation on the Kali Linux machine “C:\windows\system32\elastic-agent-8.15.3+build202411051926-windowsx86_64\elastic-agent.exe install --url=https://2190f4389f3d4ee381ad9dfe7450a096.fleet.uscentral1.gcp.cloud.es.io:443 –enrollment-token=SmlUODJaSUJhWWpQdHdUQkdOc206ei13Q1AwcGNSNkdWcjBNVEU3RDVTdw==”

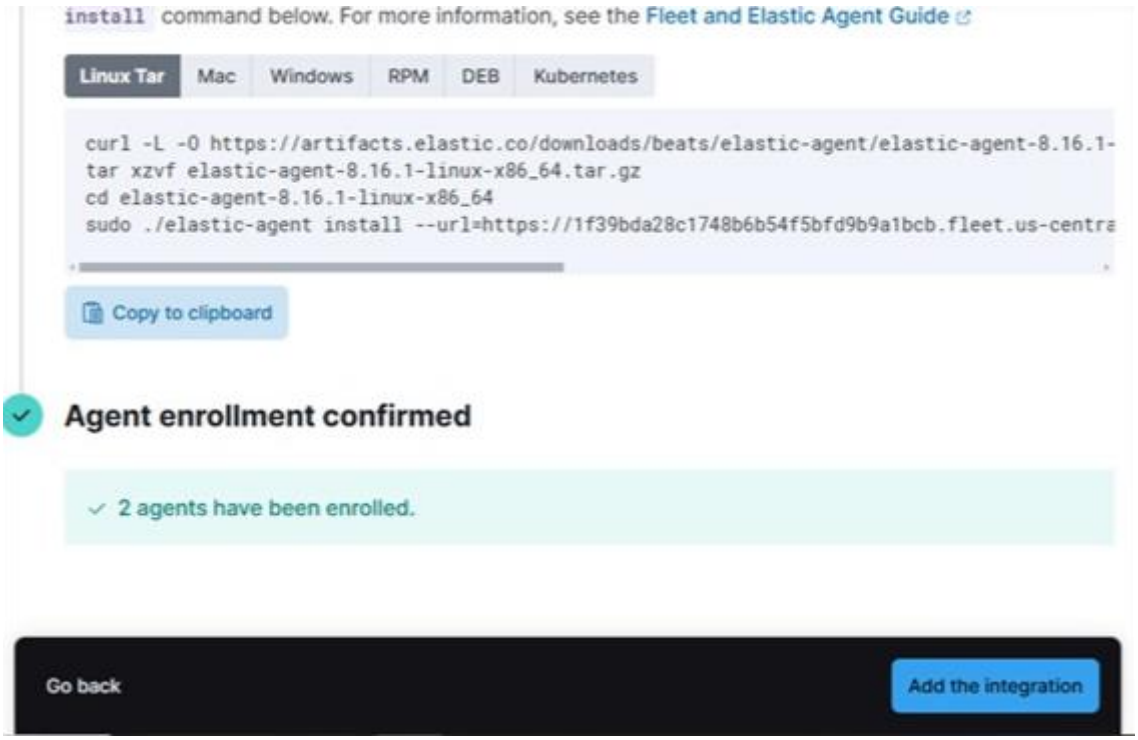
When running the command, it shows us an installation confirmation message, we enter the “Y” and “Enter” keys.

```
PS C:\> C:\windows\system32\elastic-agent-8.15.3+build202411051926-windows-x86_64\elastic-agent.exe install --url=https://2190f4389f3d4ee381ad9dfe7450a096.fleet.us
-central1.gcp.cloud.es.io:443 --enrollment-token=SmlUODJaSUJhWWpQdHdUQkdOc206ei13Q1AwcGNSNkdWcjBNVEU3RDVTdw==
Elastic Agent will be installed at C:\Program Files\Elastic\Agent and will run as a service. Do you want to continue? [Y/n]:y
[== ] Service Started  [8s] Elastic Agent successfully installed, starting enrollment.
[== ] Waiting For Enroll...  [8s]
```

At the end of the installation, we get the following message on the screen

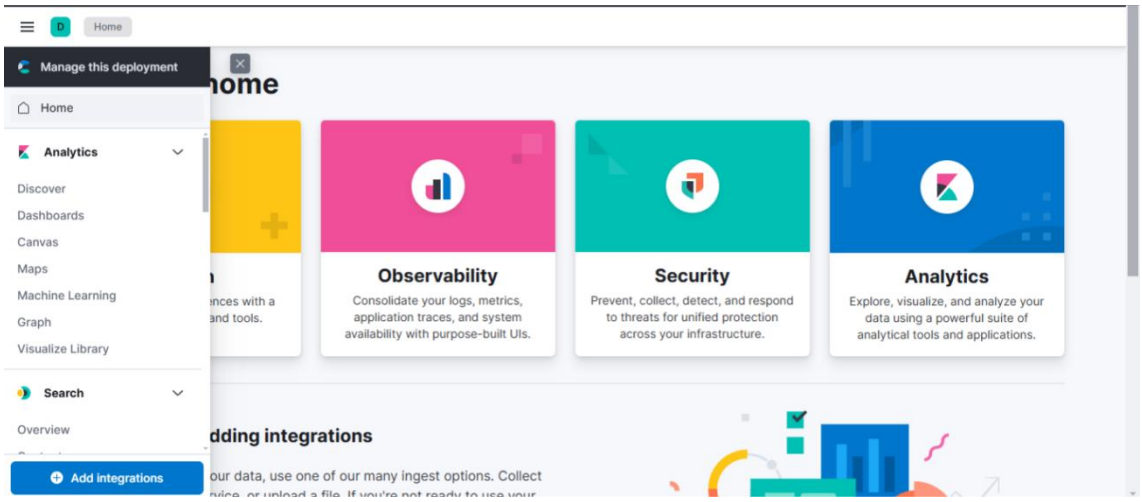
```
PS C:\> C:\windows\system32\elastic-agent-8.15.3+build202411051926-windows-x86_64\elastic-agent.exe install --url=https://2190f4389f3d4ee381ad9dfe7450a096.fleet.us
-central1.gcp.cloud.es.io:443 --enrollment-token=SmlUODJaSUJhMWpQdHdUQkdOc206eiI3Q1AwcGNSNkdWcjBNVEU3RDVTdw==
Elastic Agent will be installed at C:\Program Files\Elastic\Agent and will run as a service. Do you want to continue? [Y/n]:y
[=== ] Service Started  [8s] Elastic Agent successfully installed, starting enrollment.
[ ===] Waiting For Enroll...  [9s] {"log.level":"info","@timestamp":"2024-11-06T22:30:46.314-0500","log.origin":{"function":"github.com/elastic/elastic-agent/inter
nal/pkg/agent/cmd.(*enrollCmd).enrollWithBackoff","file.name":"cmd/enroll_cmd.go","file.line":518},"message":"Starting enrollment to URL: https://2190f4389f3d4ee38
1ad9dfe7450a096.fleet.us-central1.gcp.cloud.es.io:443/","ecs.version":"1.6.0"}
[==  ] Waiting For Enroll...  [18s] {"log.level":"info","@timestamp":"2024-11-06T22:30:55.396-0500","log.origin":{"function":"github.com/elastic/elastic-agent/inte
rnal/pkg/agent/cmd.(*enrollCmd).daemonReloadWithBackoff","file.name":"cmd/enroll_cmd.go","file.line":481},"message":"Restarting agent daemon, attempt 0","ecs.versi
on":"1.6.0"}
{"log.level":"info","@timestamp":"2024-11-06T22:30:55.425-0500","log.origin":{"function":"github.com/elastic/elastic-agent/internal/pkg/agent/cmd.(*enrollCmd).Exec
ute","file.name":"cmd/enroll_cmd.go","file.line":299},"message":"Successfully triggered restart on running Elastic Agent.","ecs.version":"1.6.0"}
Successfully enrolled the Elastic Agent.
[   =] Done  [18s]
Elastic Agent has been successfully installed.
PS C:\>
```

Now that we have both agents enrolled, we check the following notification in the Elastic integrator and click on the “Add the integrations” button:

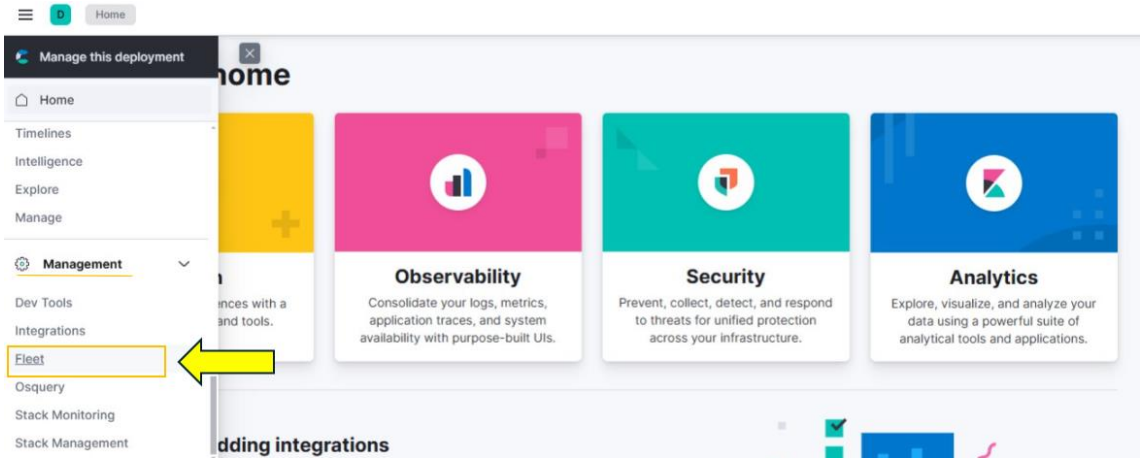


Validating that machine logs are being received correctly

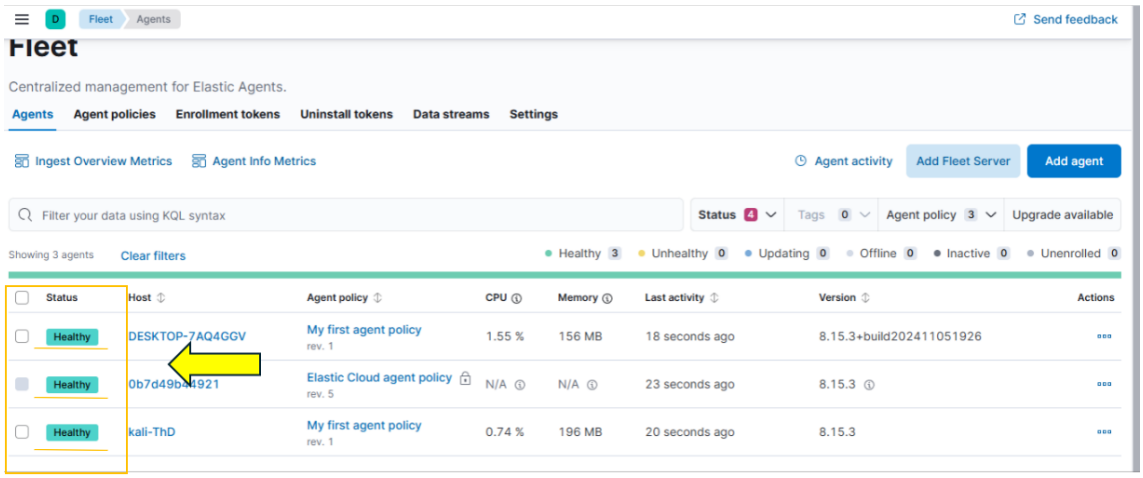
We enter the Elastic main page and display the menu on the left side of the screen.



We look for the button *fleet* in the menu located in the Management submenu



Here we find our machines and previously created projects



You can see the agents that are active and "healthy" in the status box. If they are offline and grayed out, you should check the corresponding machines to see if the agent is running.

Now in the submenu we click on “Data streams” and we will be able to observe the *fleet*of the machines. This means that the machine agents are communicating properly with our project in*Elastics*.

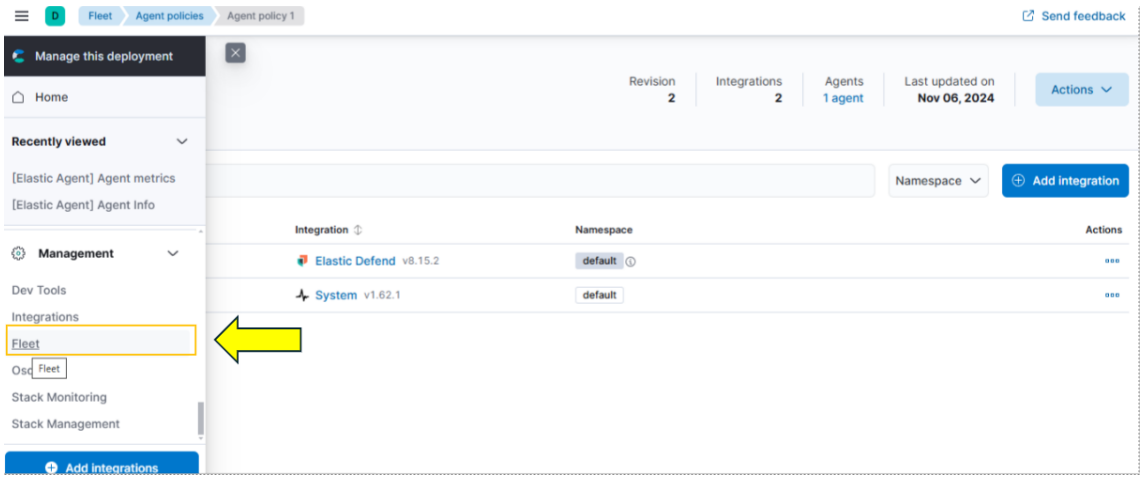
Uninstall Elastic on virtual machines

You can uninstall the installed software at any time as follows. For both Windows and Linux, open the command prompt and paste one of the following links, depending on the operating system.

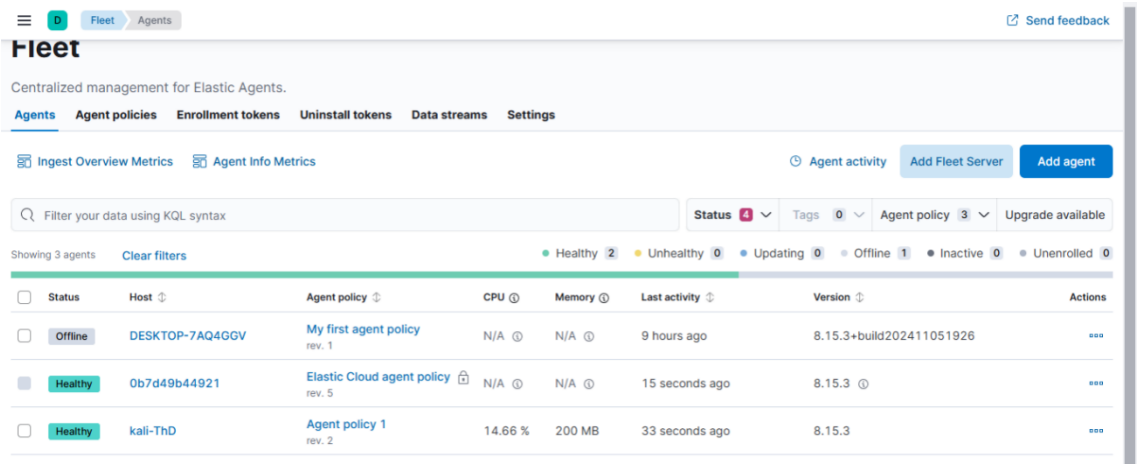
- Windows: C:\"Program Files"\Elastic\Agent\elastic-agent.exe uninstall
- Linux: -> sudo /opt/Elastic/Agent/elastic-agent uninstall

Installation Problems

If the logs aren't showing in the Elastic console, you'll need to check the Fleet menu to make sure the agents have the correct integrations assigned.



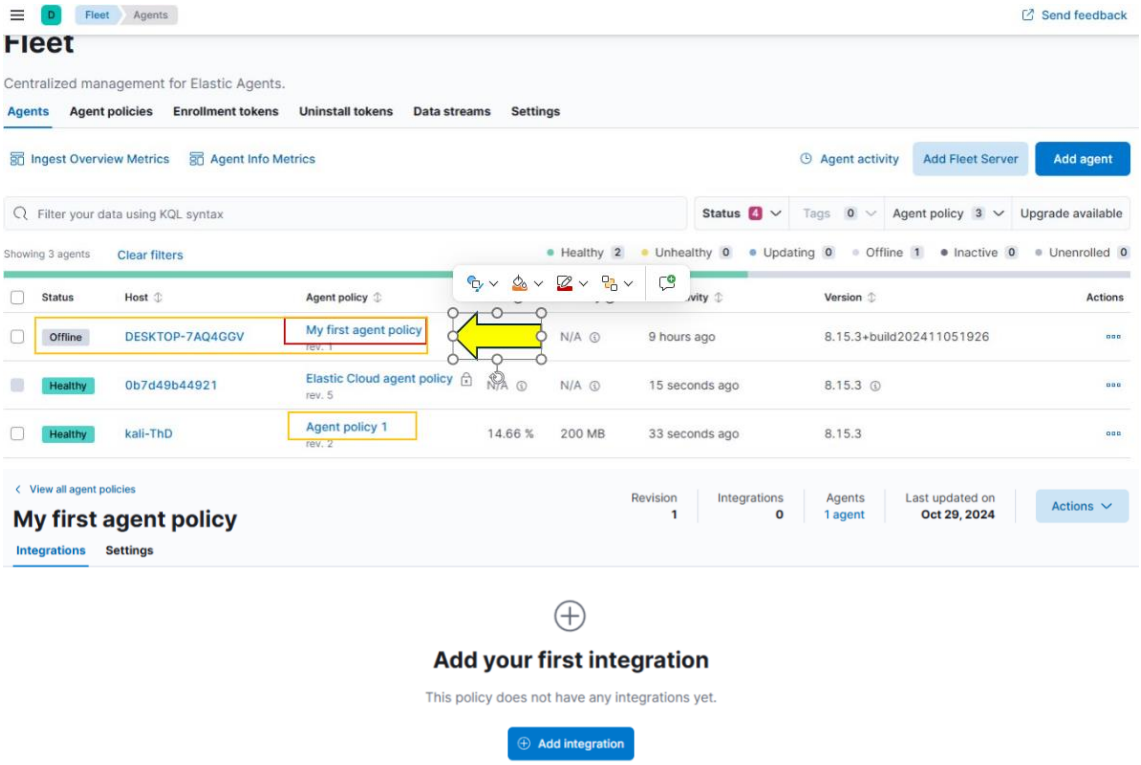
In this window the agents and policies will be displayed



In this case our integration policy is called “Agent Policy 1”, we can see that for the Kali machine it is correctly assigned, however, for Windows 10 it is not.

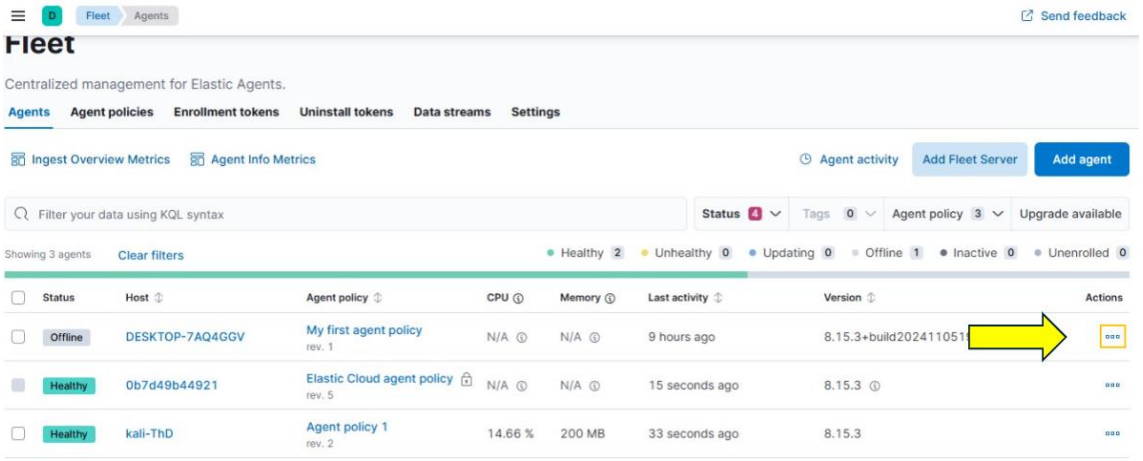
We locate the “Agent Policy” column and click on the box corresponding to the Windows machine, which in this case has the name “Host DESKTOP-7AQ4GGV"

In this case we click on “My first agent policy”

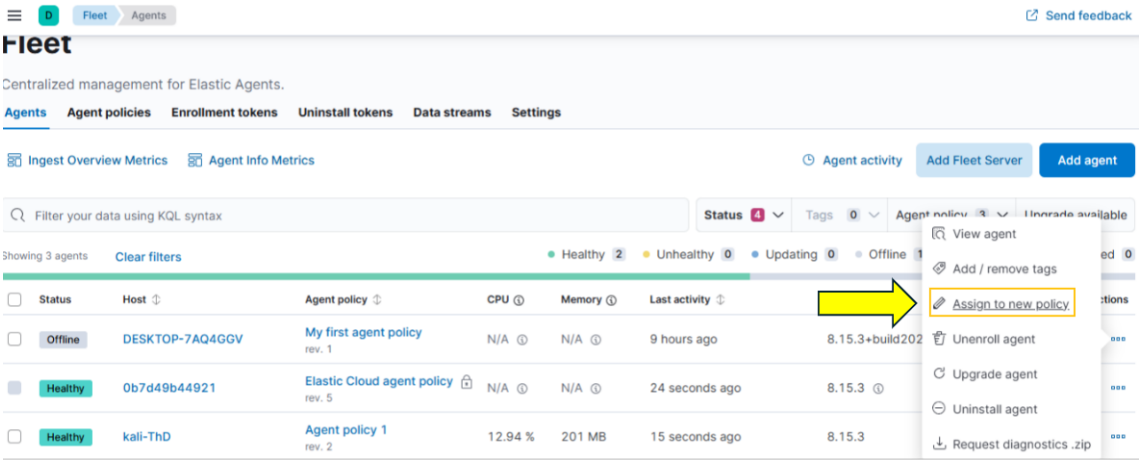


We can see that it has no integrations.

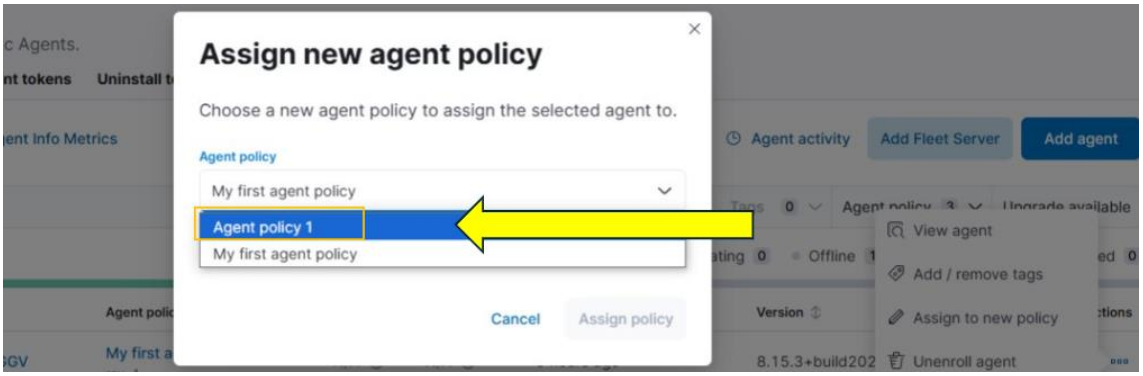
To change it to the correct integration policy, go back to the previous page “view all agent policies.” In the Windows machine row, locate the three-dot menu on the right.

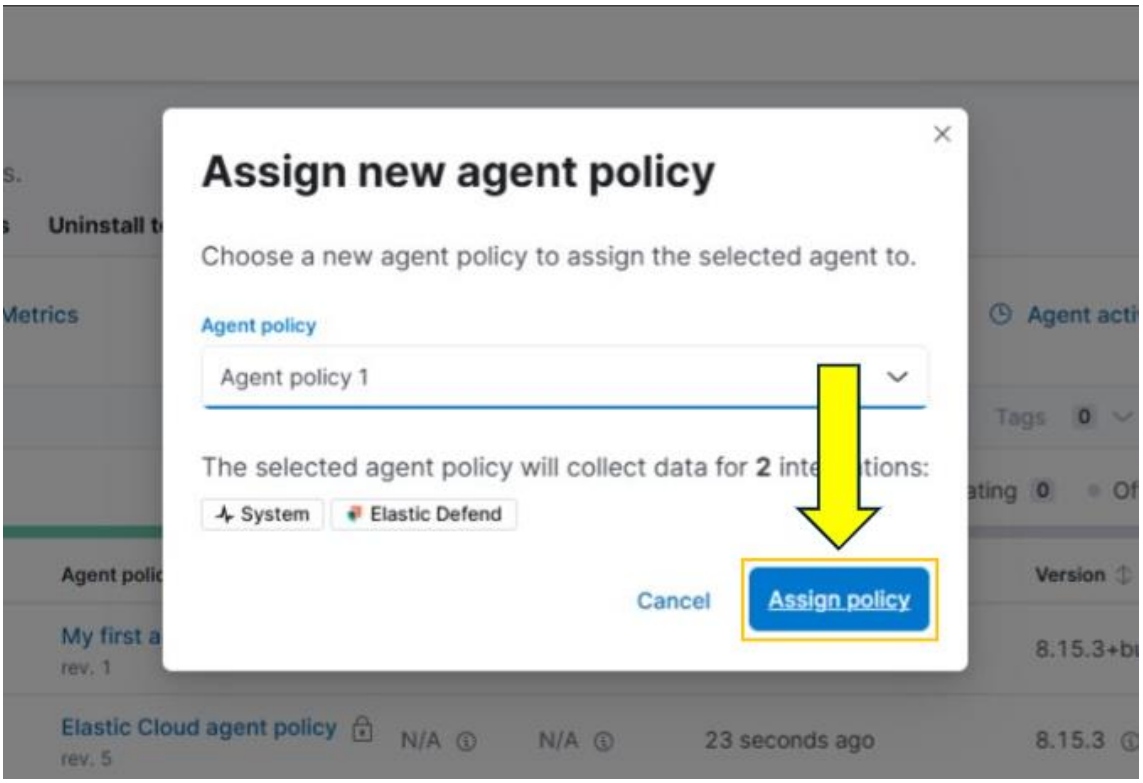


We select the option “Assign to new policy” A new window will be displayed where we select the correct policy that contains our integration with *elastic defender.*



We click on “Assign policy”





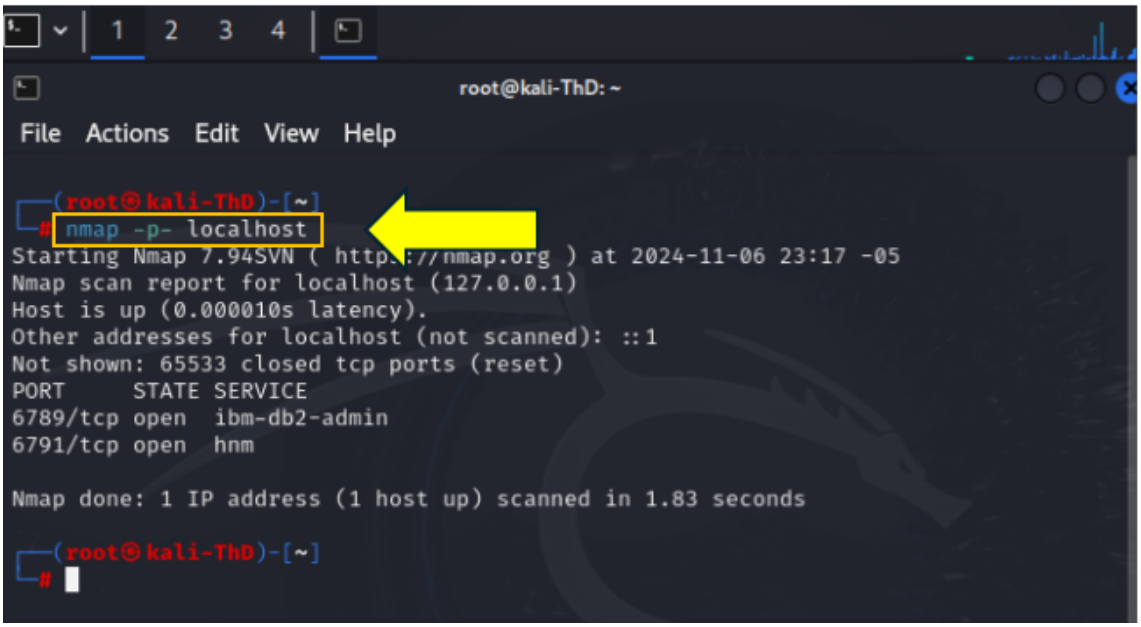
For the changes to take effect, we turn on the Windows 10 virtual machine and wait for the service to update.

Getting Started with the Product - Generating Events in Kali and Windows

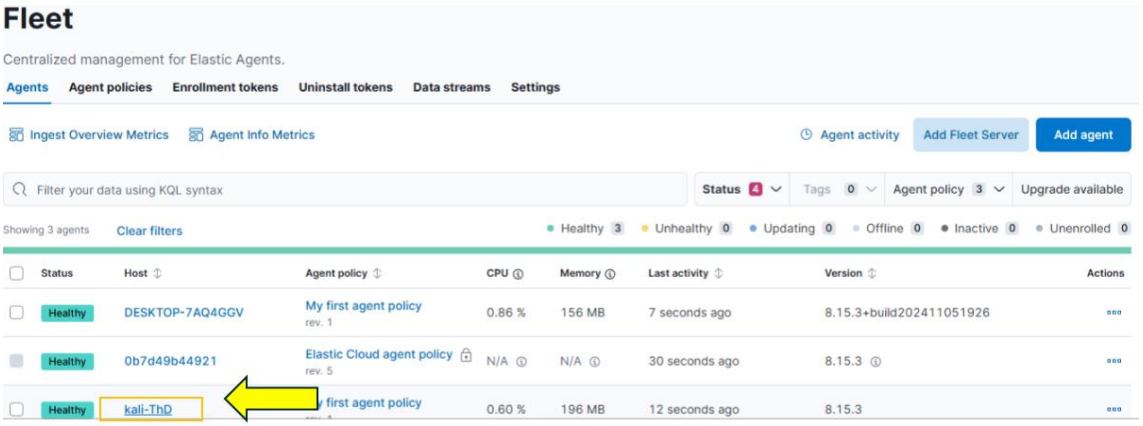
Now that we have the agents running successfully, we proceed to generate events for each one so that they are reflected in the Elastic console.

Generating events in Kali and viewing them in the log console elastic

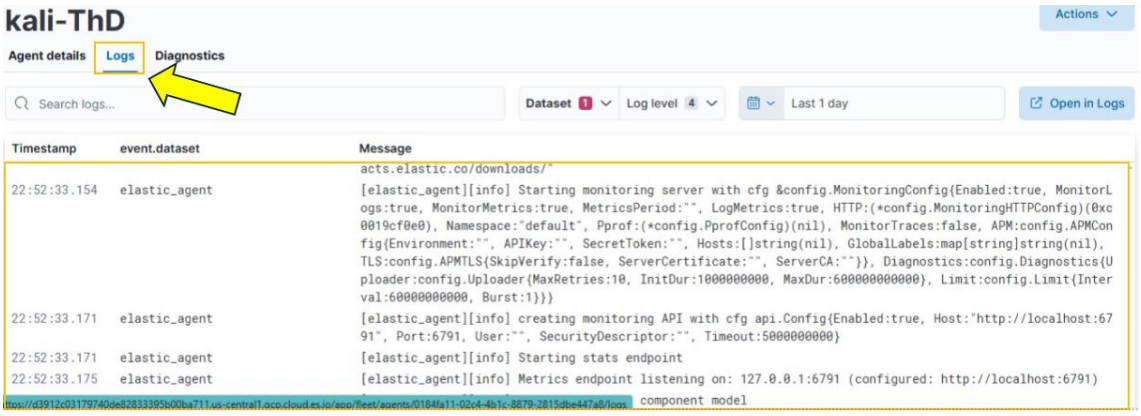
For this activity, we'll be using Kali's NMAP tool. In a Kali command prompt, enter the following command: "nmap -p-localhost"



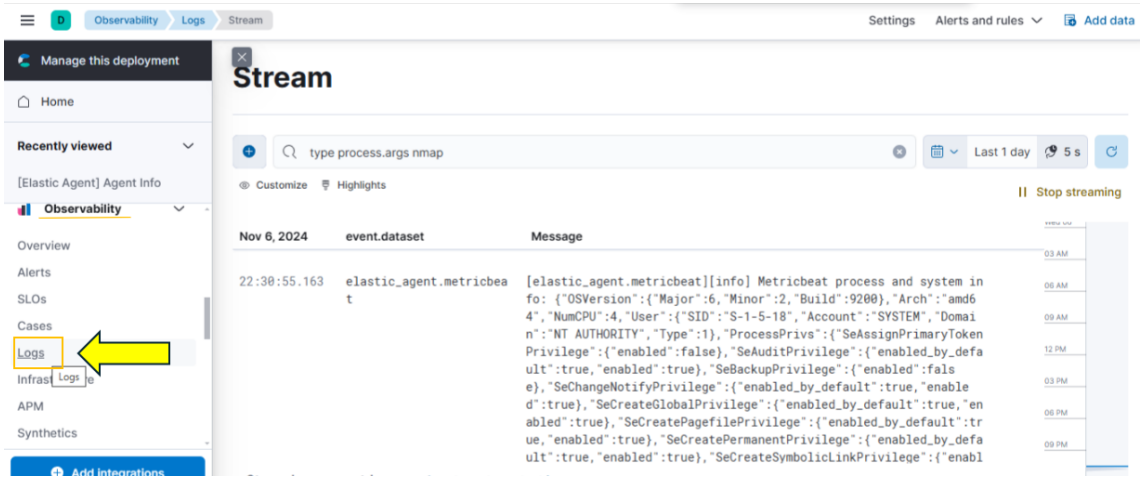
To validate that the command is in the events of e*elastic*, we go back to the menu page*fleet*and we enter the Kali machine.



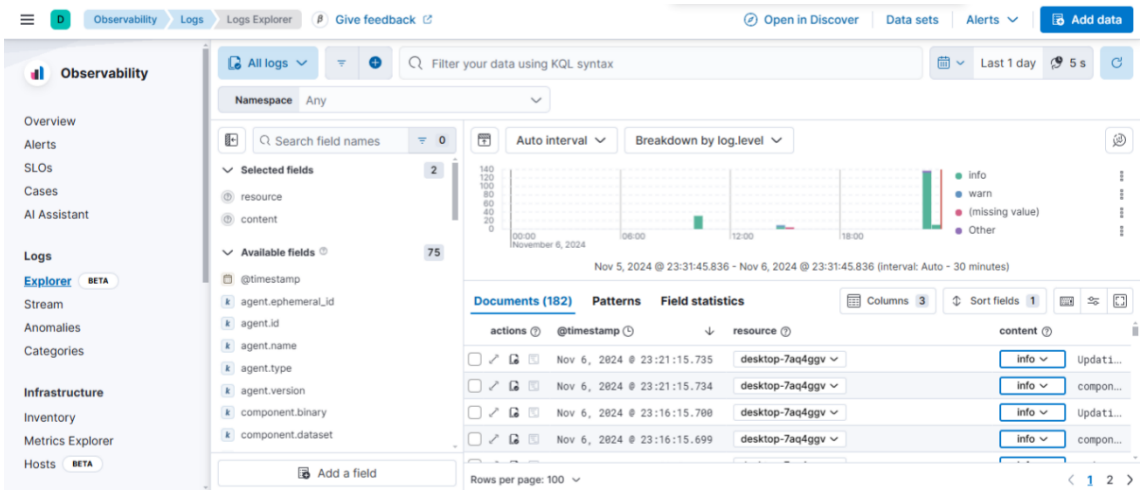
In the machine window we locate the tab in the submenu*logs*, we click on it and we will be able to view the*logs*of the machine.



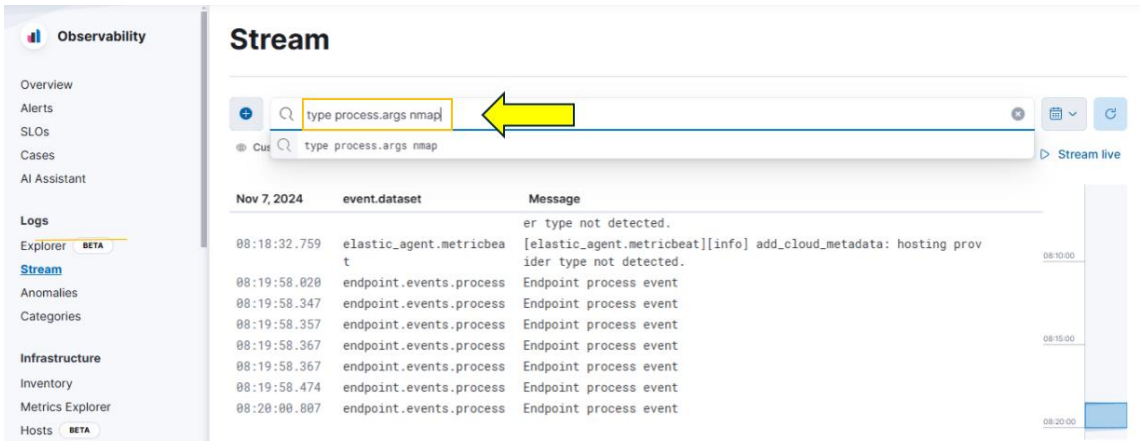
To check that the console is working*Elastic*is receiving the events from the agents, we display the three-bar menu and look for the “Observability” submenu and click on the button *Logs*



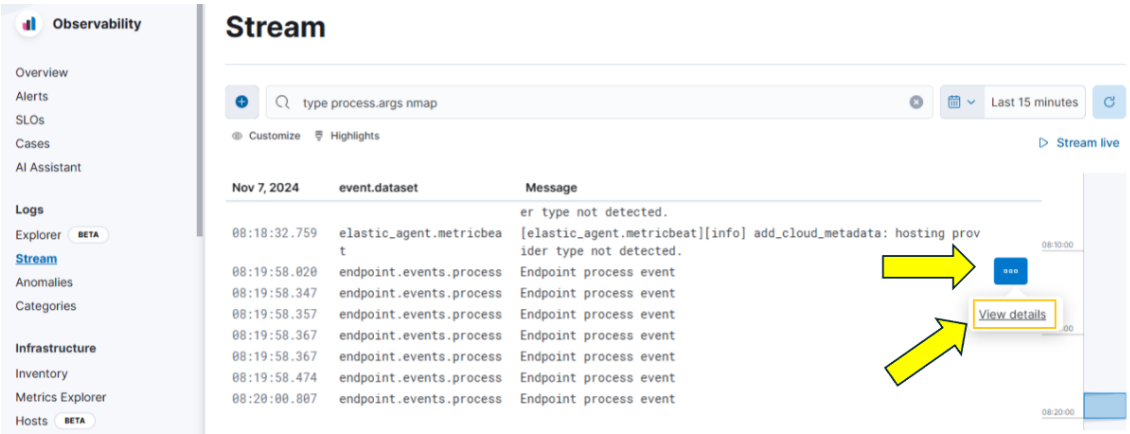
In this way we enter the log display panel



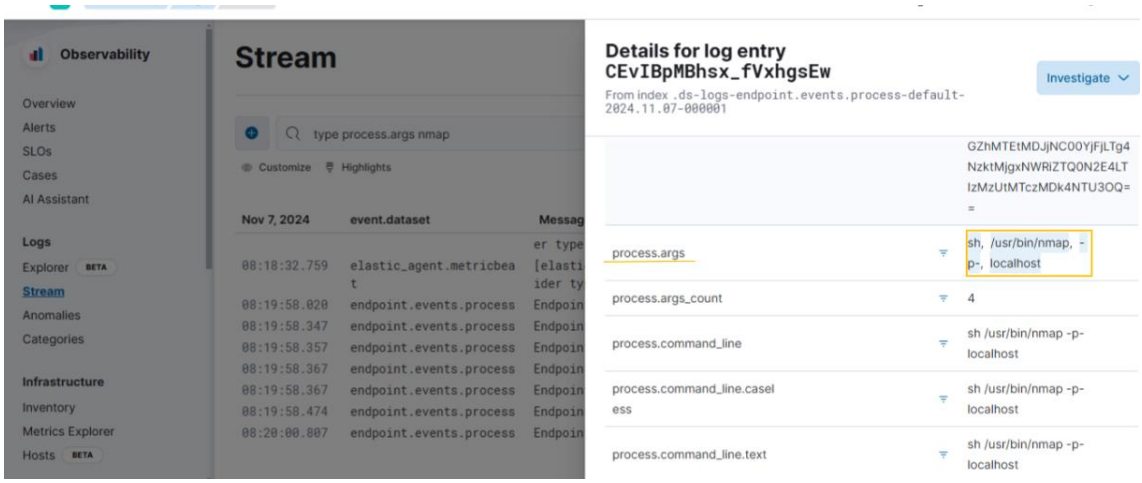
Now we proceed to filter using the search bar and paste the following command “type process.args nmap”



We can see that the filters are *logs* where there are events of *endpoint*, we locate one of them and a menu button with three dots appears and we click on "View details".



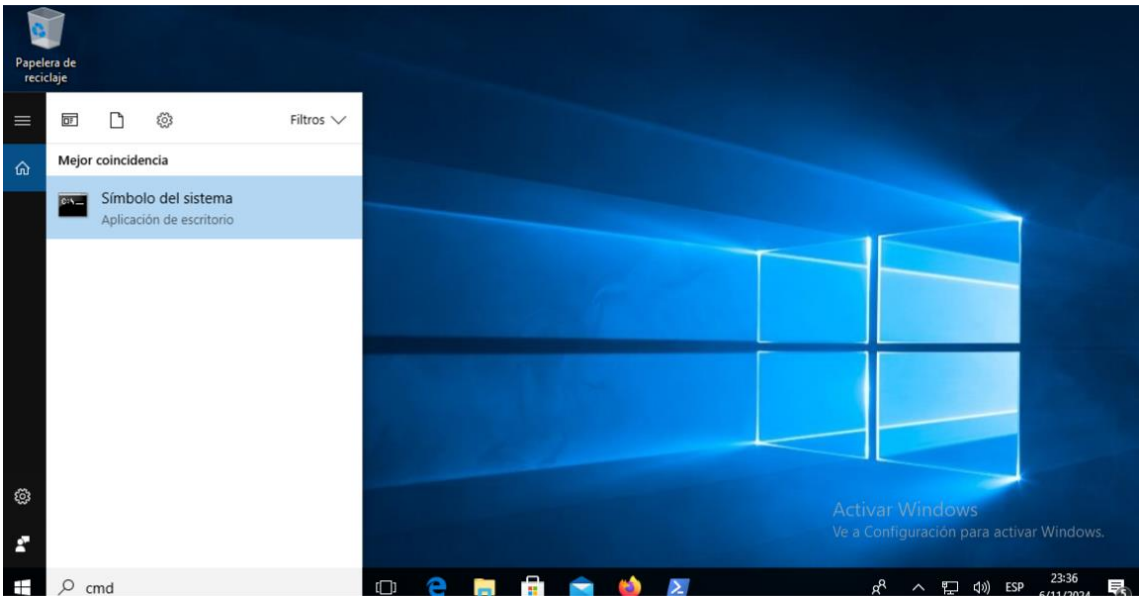
A side menu is displayed with the details of the *log*, we go down and look for the "process.args" box and we see that it contains the command that we used on the Kali machine.



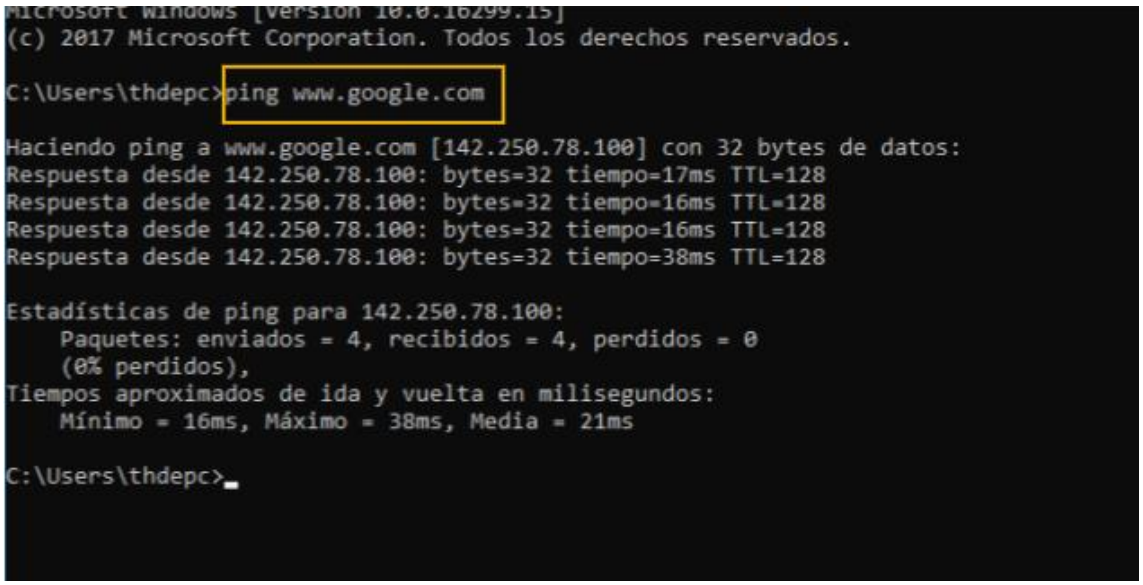
This shows us that the *logs* are received correctly on the console *elastic* through the "Elastic Defend" integration.

Generating events in Windows 10 and displaying them in the log console of elastic

On the Windows 10 machine, type the cmd command in the search bar



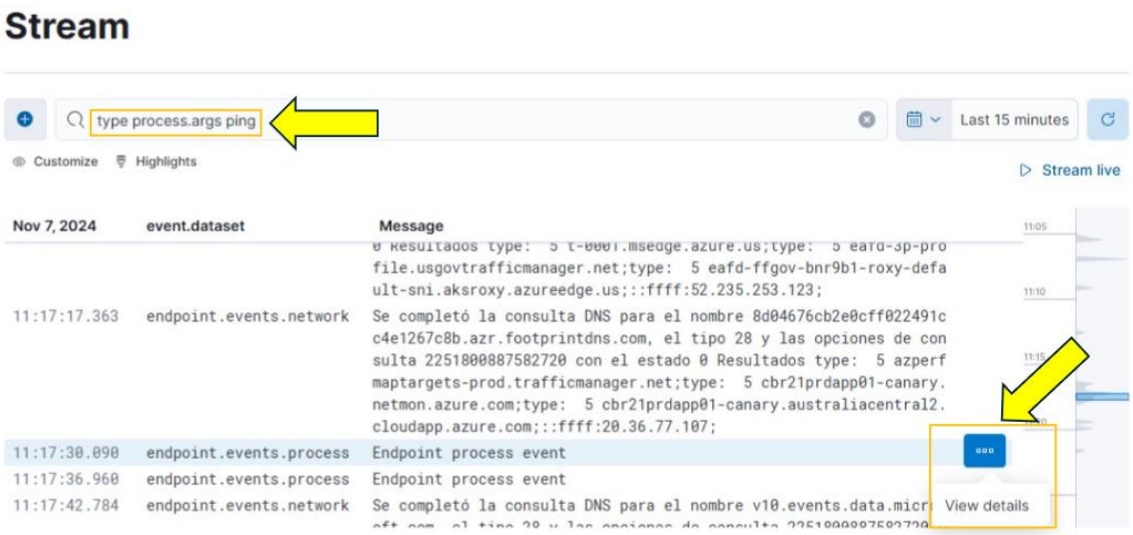
We open a window of *command prompt*. In this window we do *ping* to Google, using the command *ping* and the *url* from the page "www.google.com"



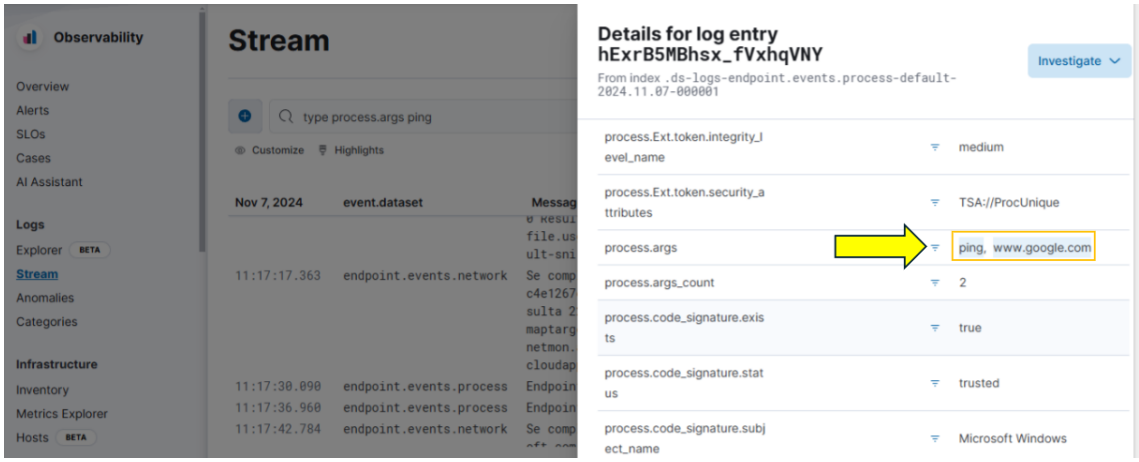
Now we go to the console *logs* of *elastic* and we filter the search with the command "type process.args ping"

Following the same methods used on the Kali machine, we display the three-point menu and click on "View details", we will find the command entered in the console. Following the same methods used on the Kali machine, we display the three-point menu and click on "View details", we will find the command entered in the console.

console

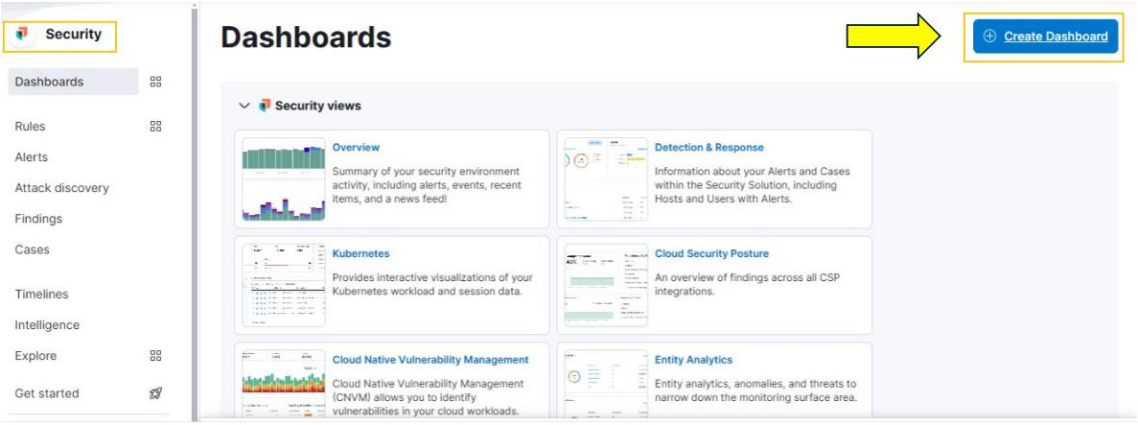


Following the same methods used on the Kali machine, we display the three-point menu and click on “View details”, we will find the entered command in the console

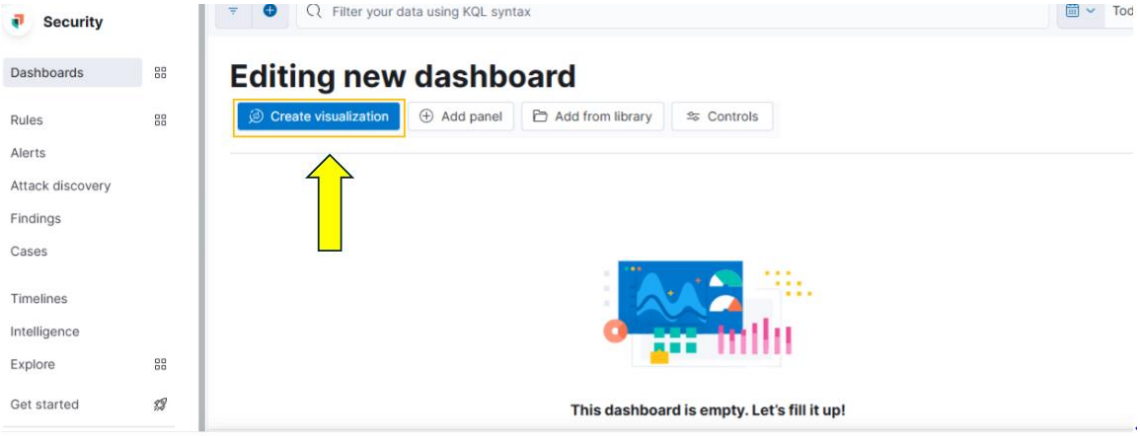


Create a dashboard to visualize events

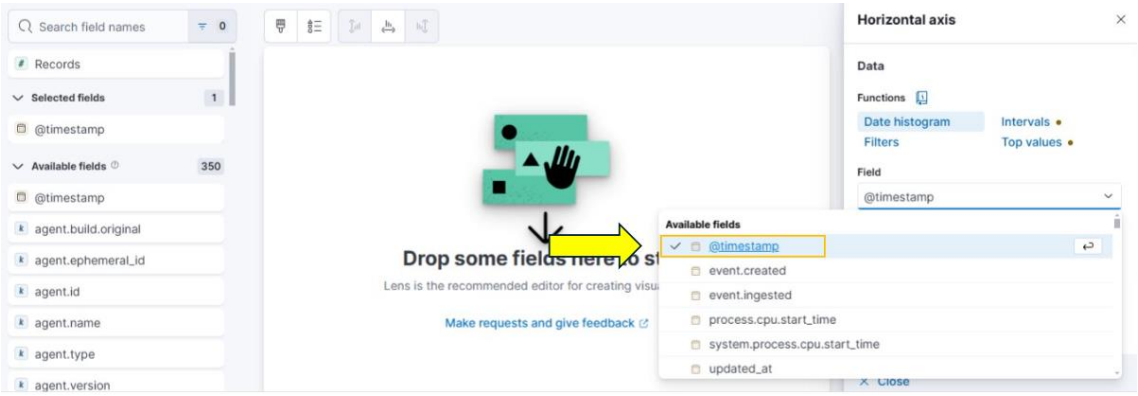
We go to the three-line menu and look for the “Security” section, in it we click on the “Dashboard” option, then we click on the blue “Create Dashboard” button that appears in the upper right corner of the screen



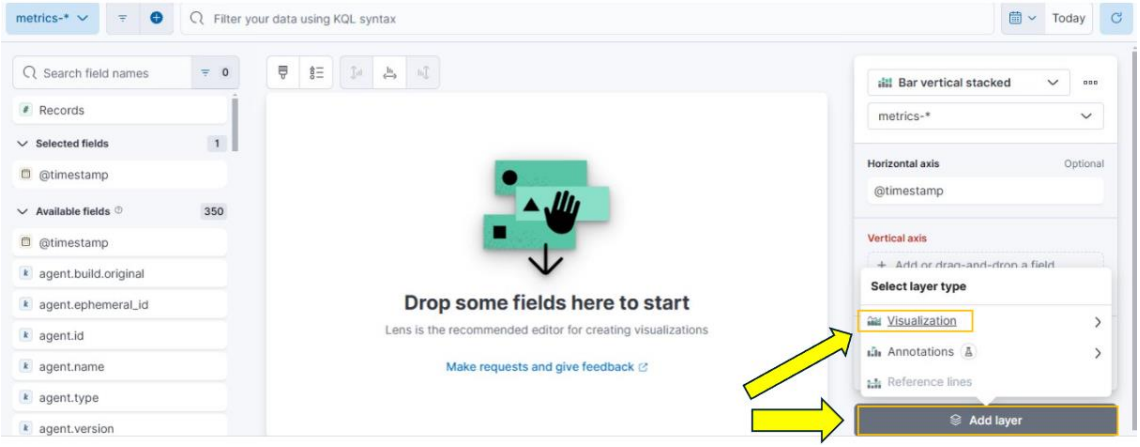
We click on “Create visualization”



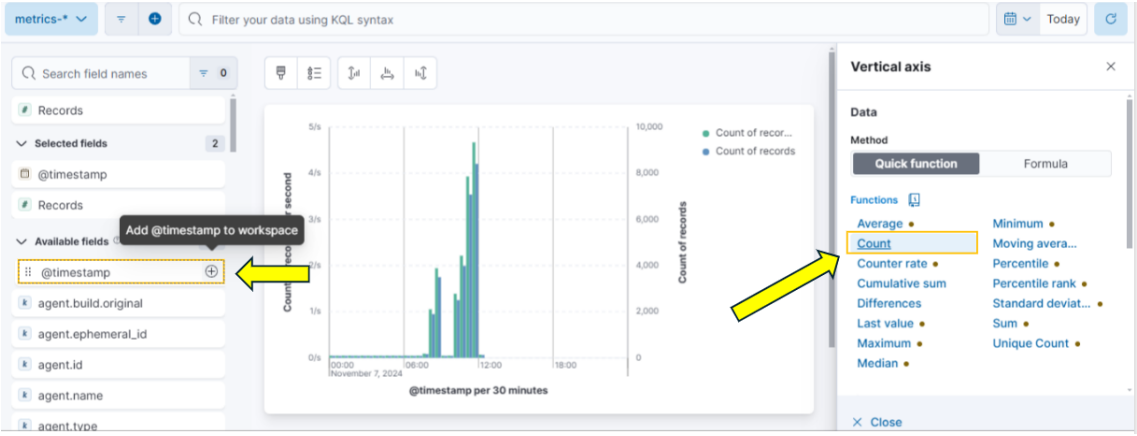
On the horizontal axis “X” of the graph we select “@timestamp”



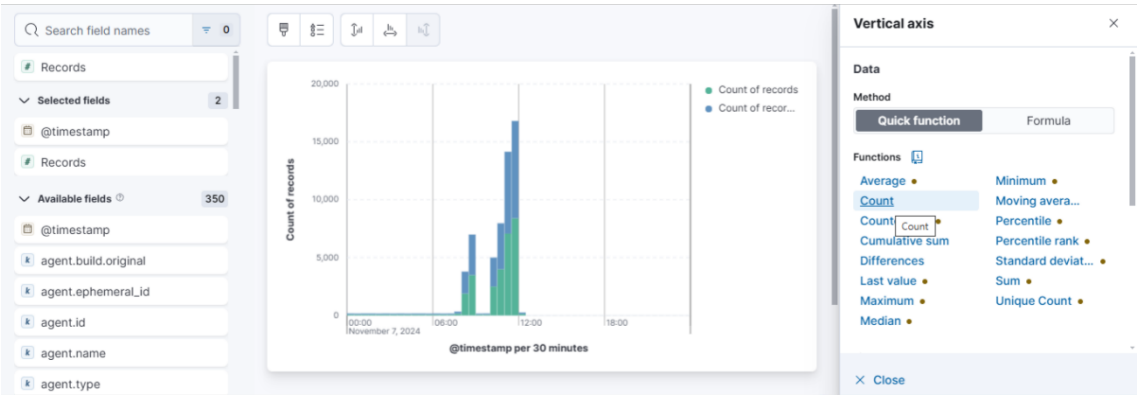
In the “add layer” button we select “visualization” For the vertical axis “Y”



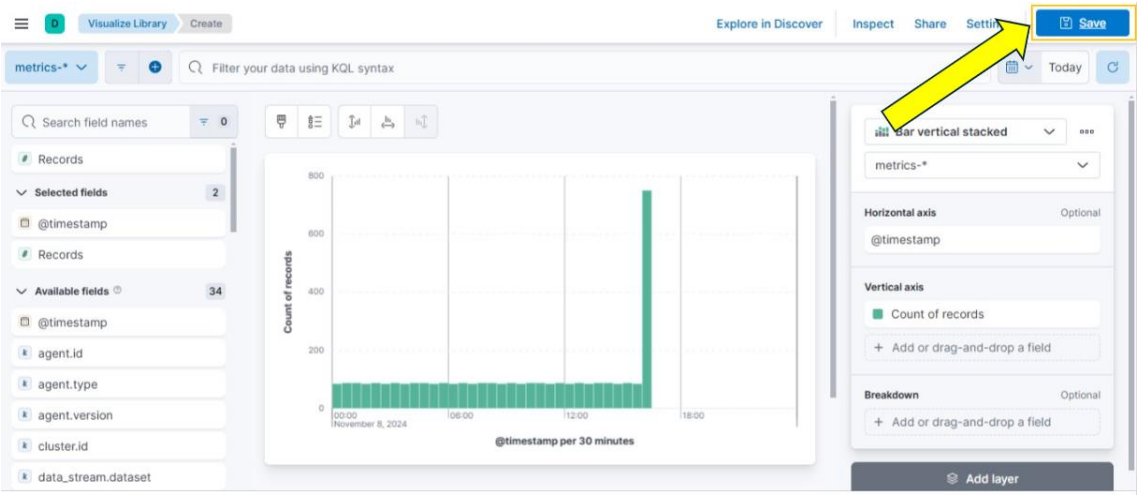
We select “Count” and in the left sidebar on “@timestamp” we click on the “+” symbol and add it to the “workspace”



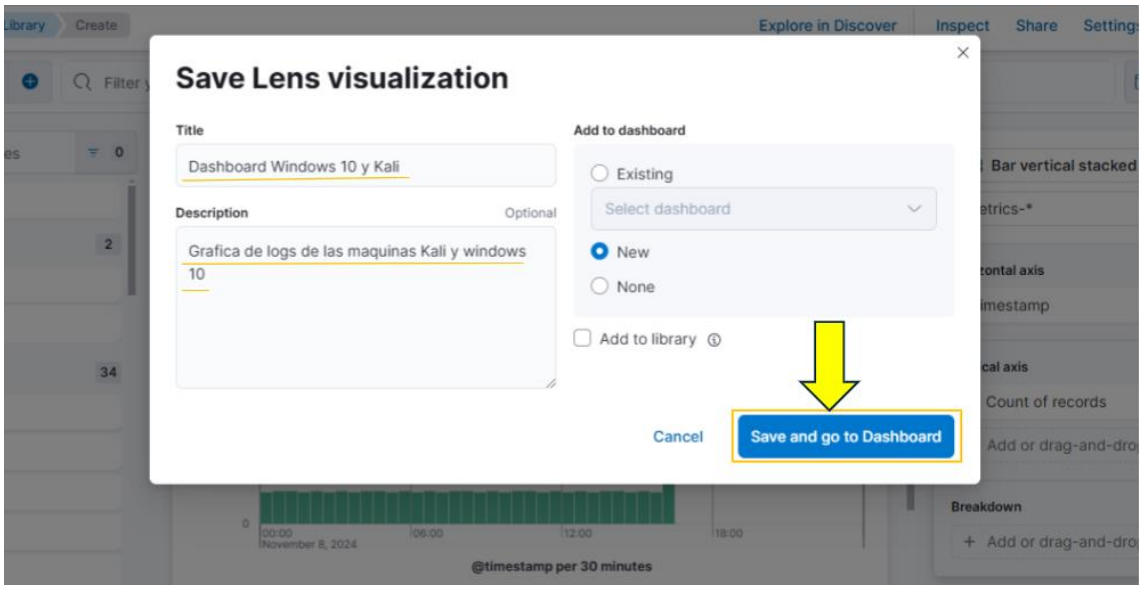
In this way we have the graph with the events



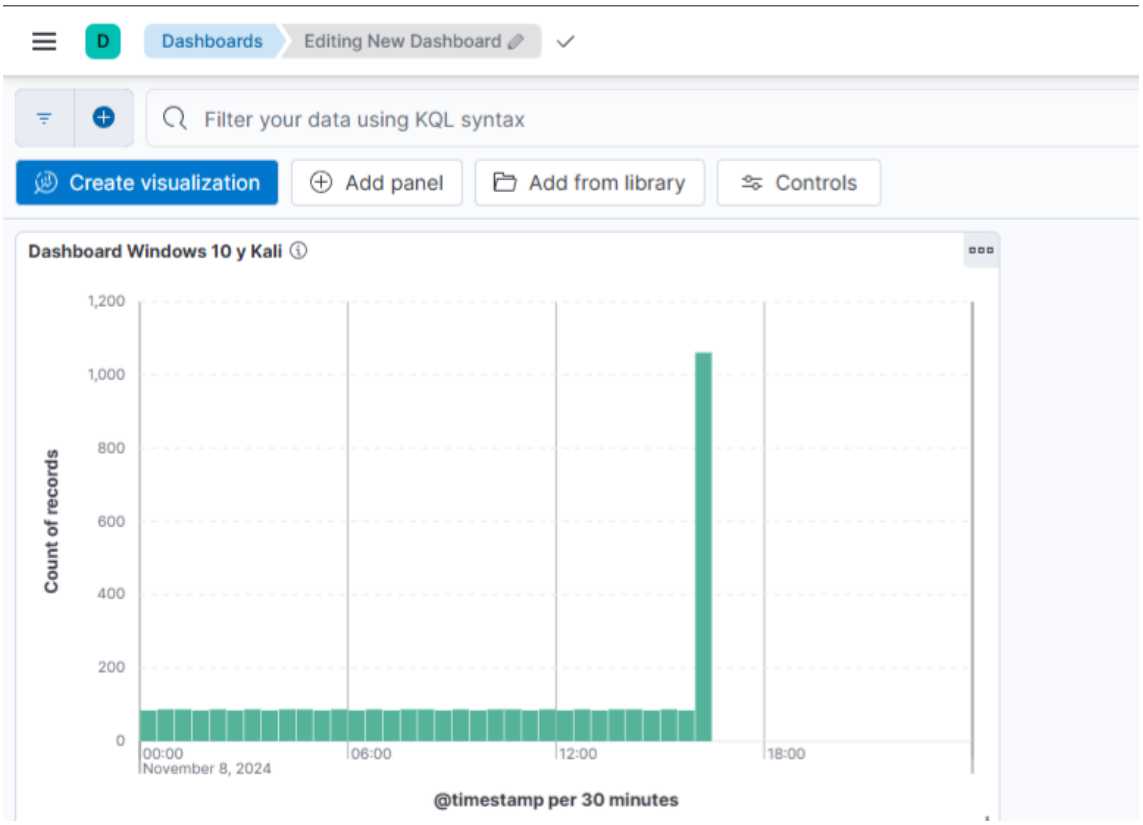
We can save the *dashboard* by clicking on the “Save” button at the top right



In the window that appears, we add the dashboard data.



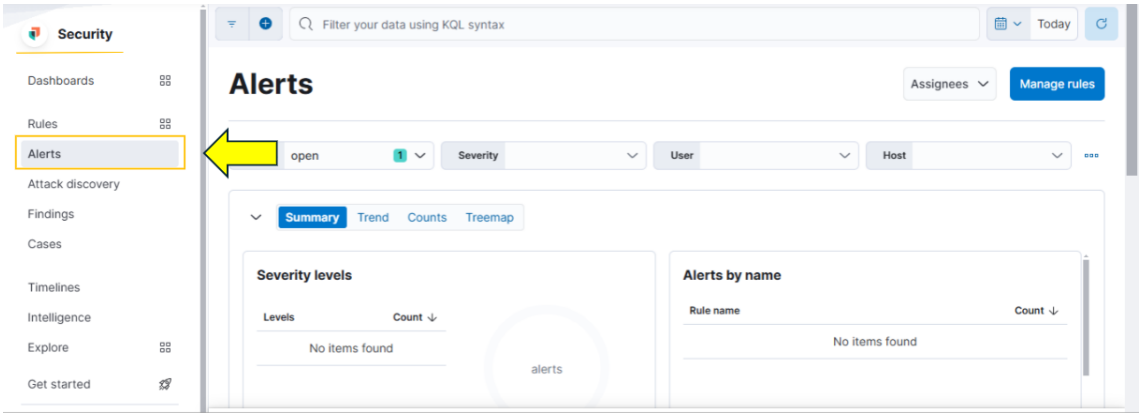
And when clicking on “Save and go to Dashboard” it shows us our *dashboard* created and configured



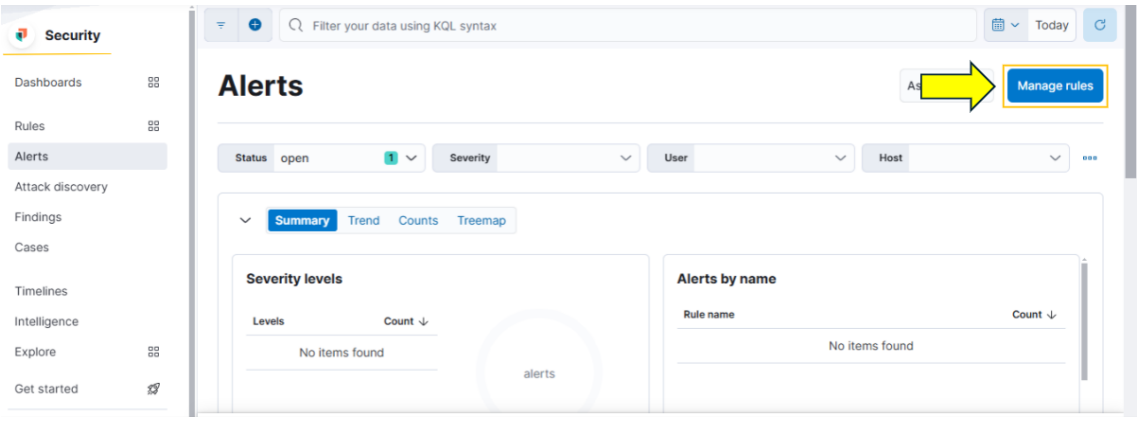
Creating alerts

Alerts allow us to detect security incidents in a SIEM environment. In this case, we will create an alert to detect Nmap on our machines.

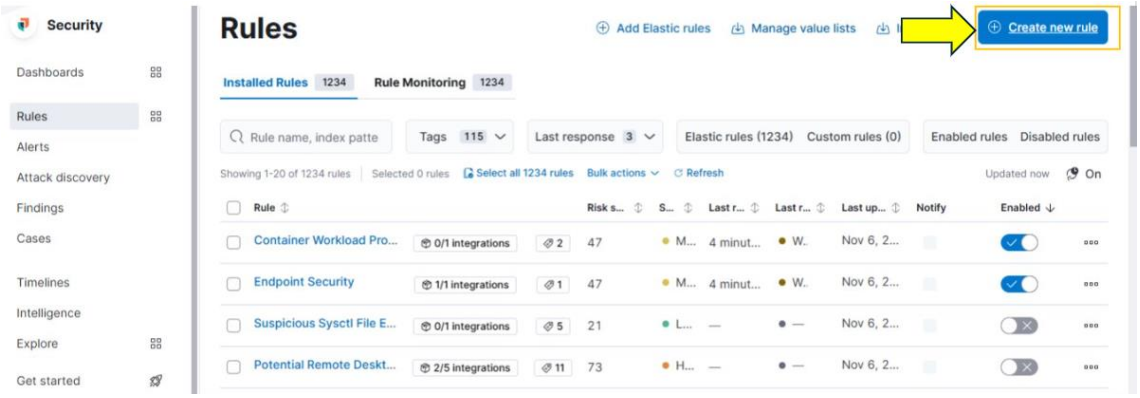
To successfully create alerts, we display the three-line menu and look for the "Security" section and click on the "Alerts" menu.



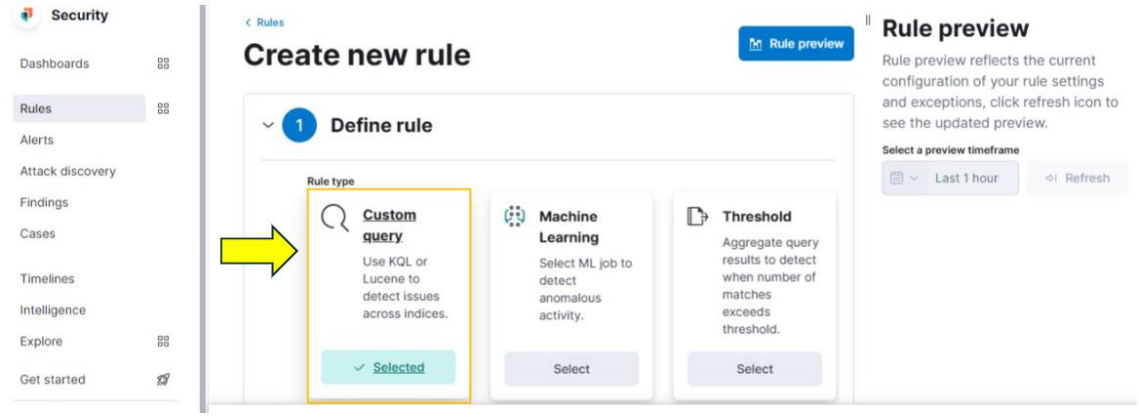
We click on the blue button that says "Manage rules"



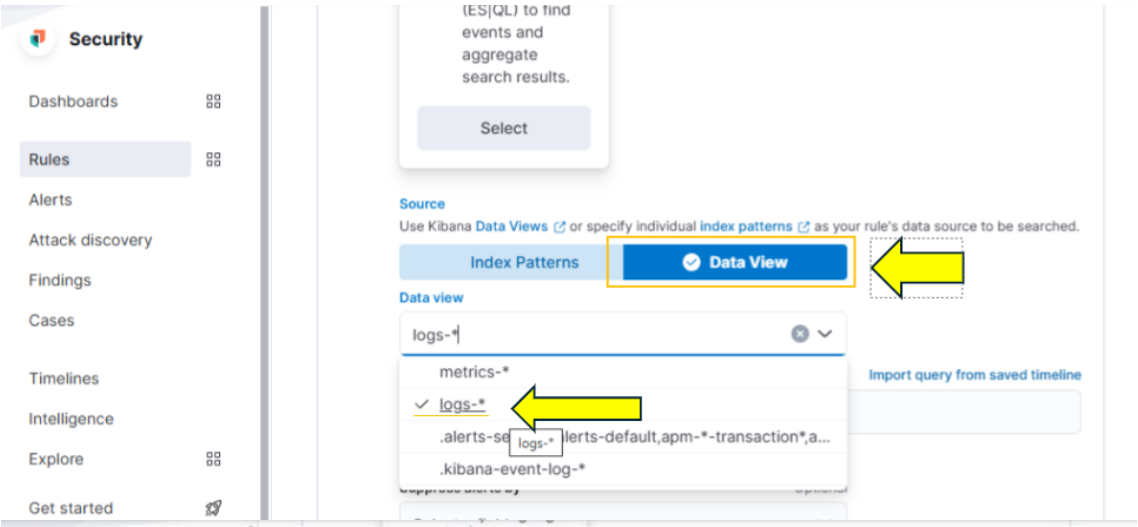
And we click on “Create new rule”

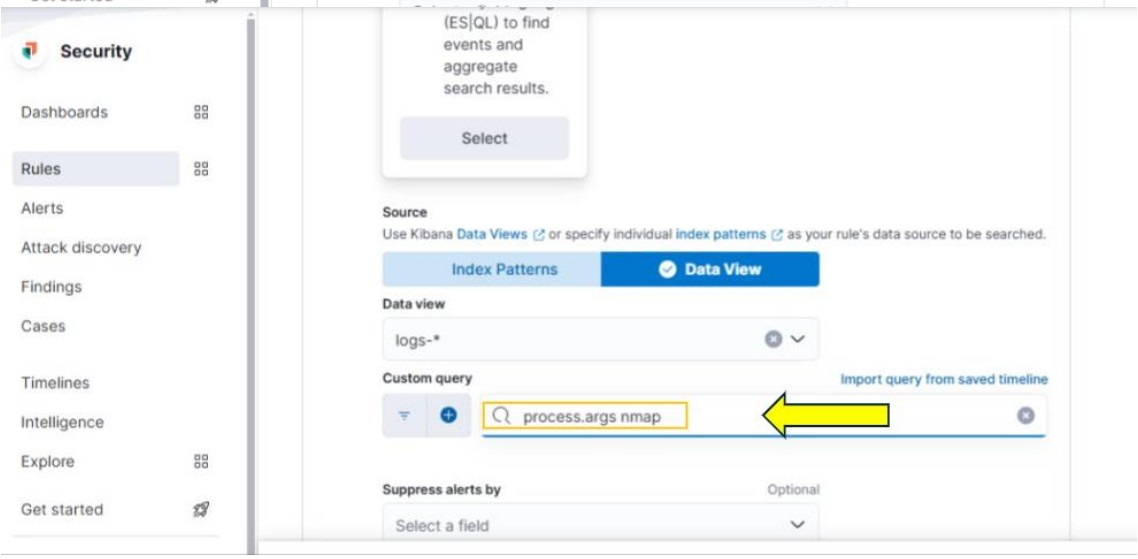


We select “Custom query”

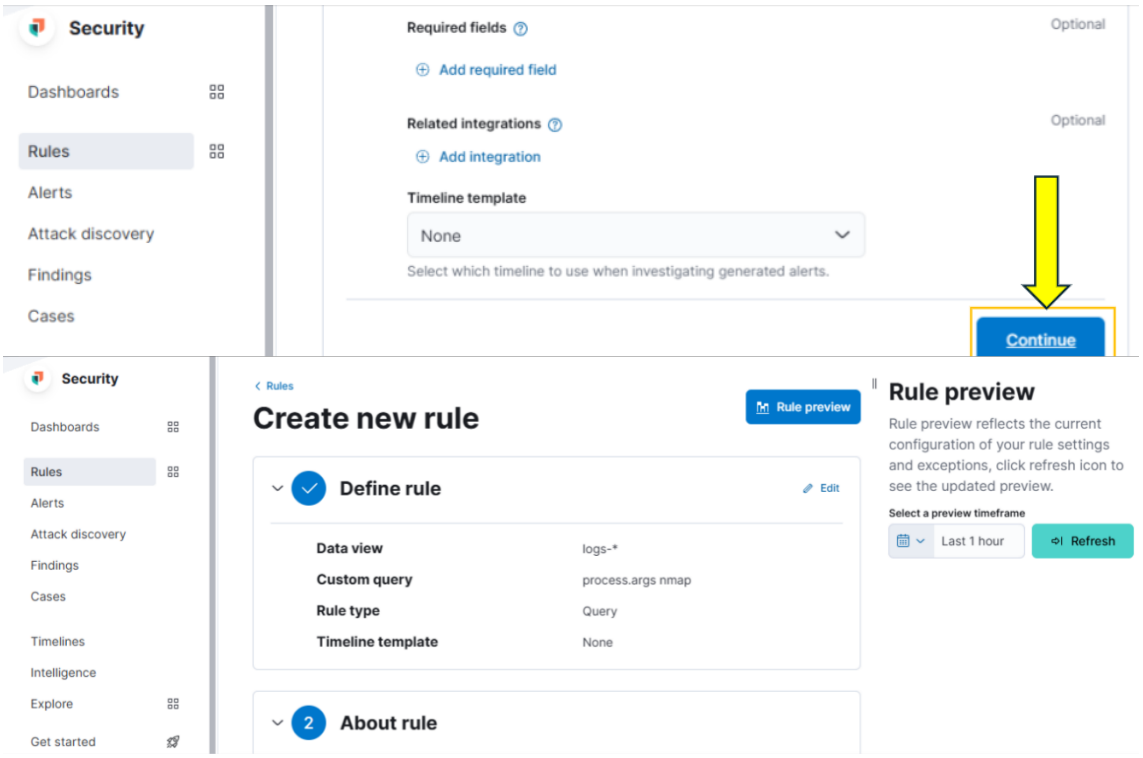


We go down and click on “Data View” and select “logs-*” then add the “process.args nmap” filter



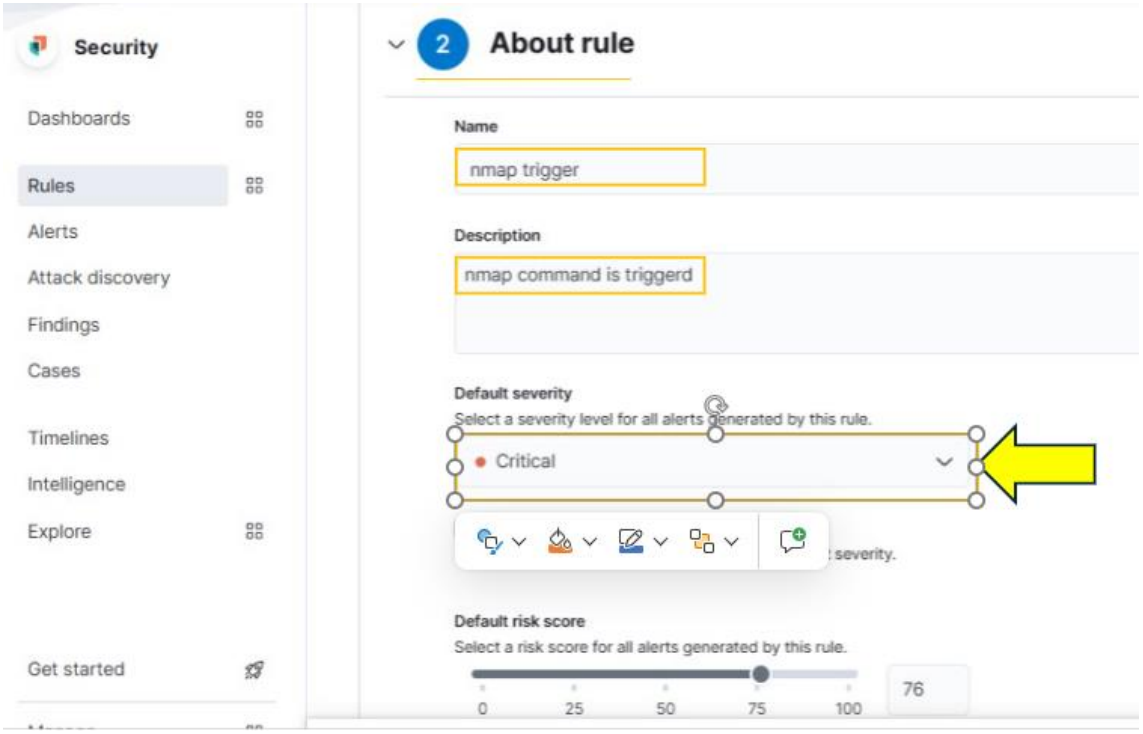


And we click on continue, which is located below

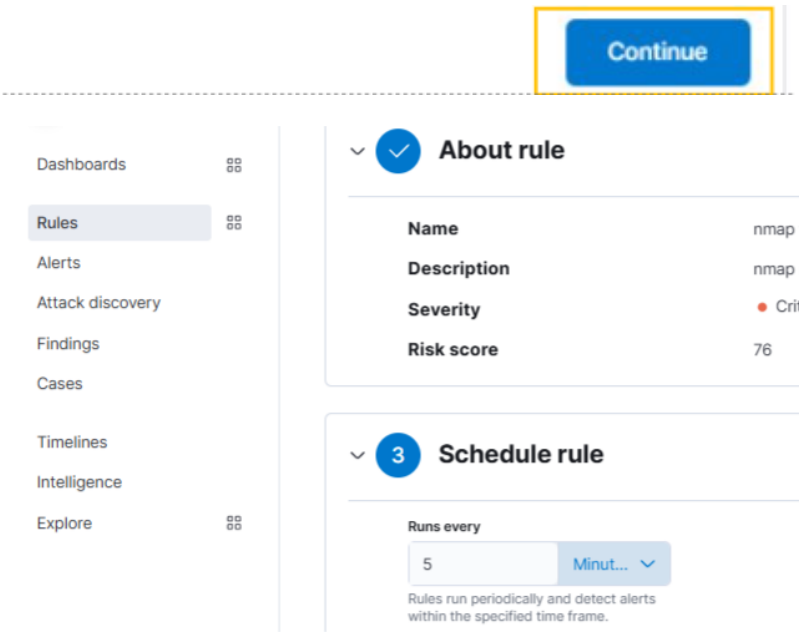


In this way we see that our rule was created.

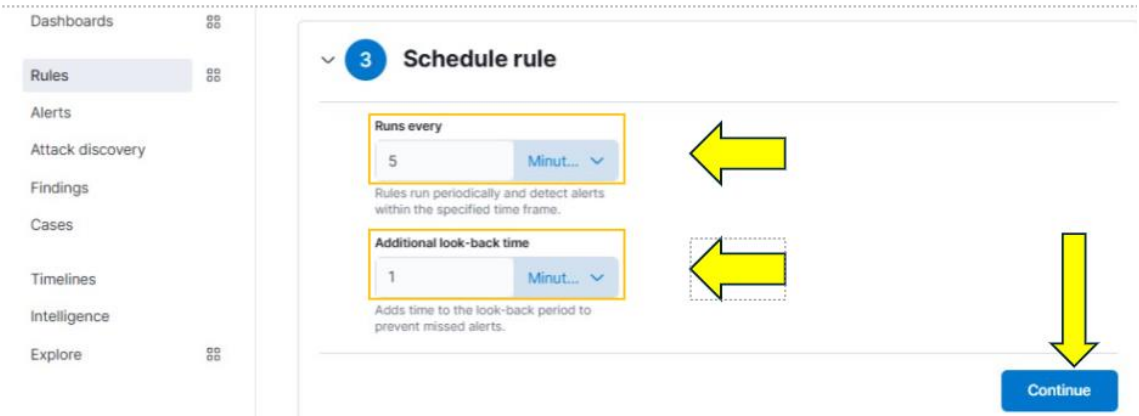
Now, in the “About rule” section we will create the *trigger* From the alert, we enter a name in “Description” and how critical we want the alert to be



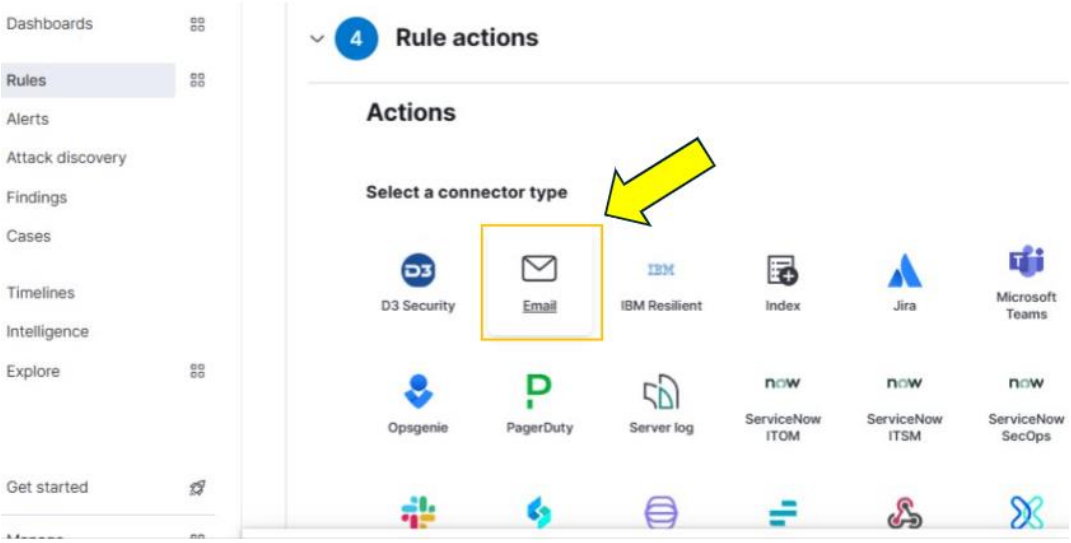
We click on continue



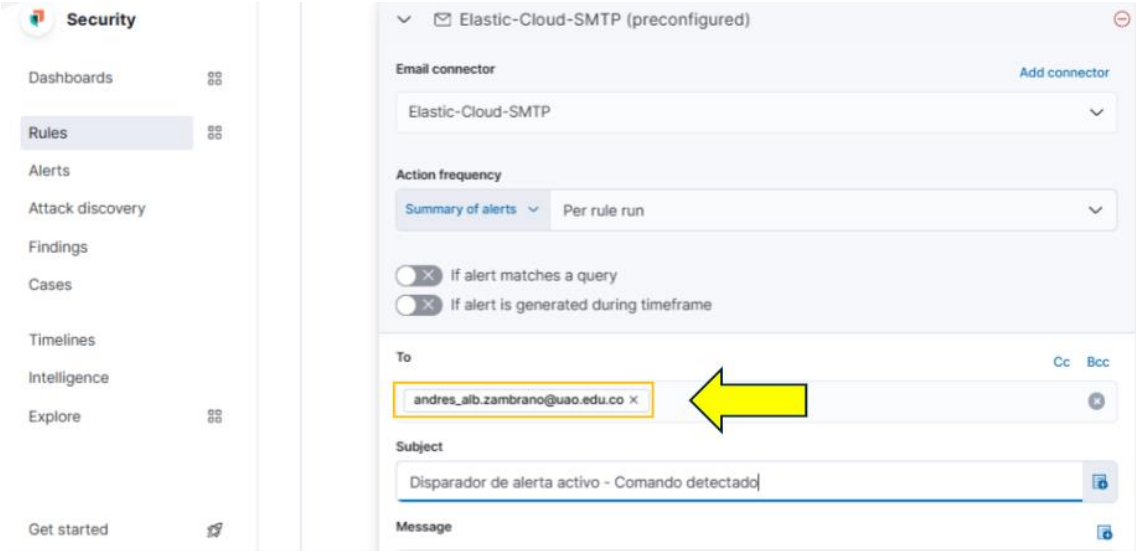
We assign the execution times and the frequency of the alert and click on continue.



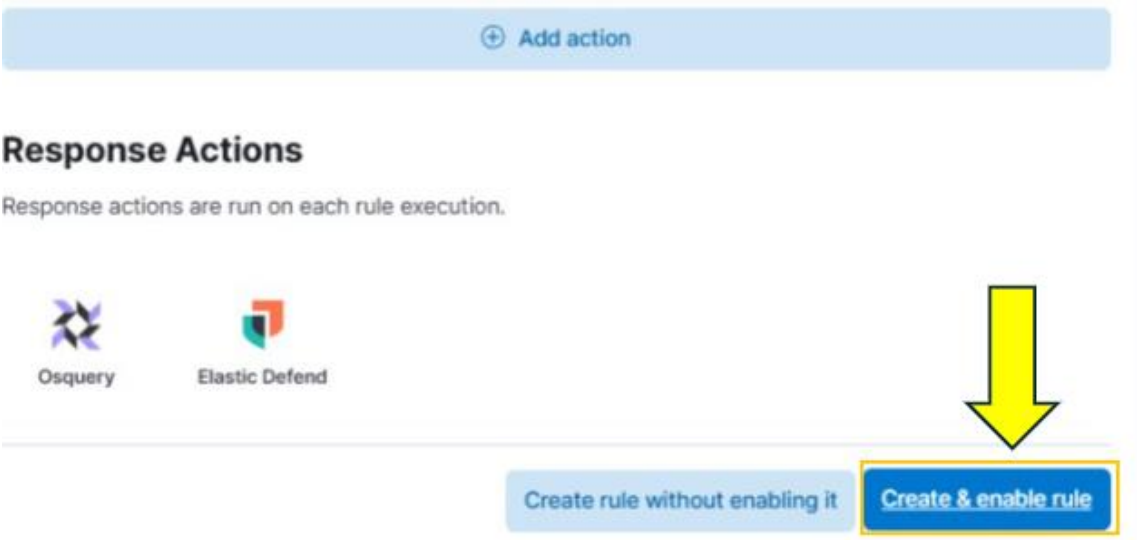
Now in the actions that the rule or alert will perform, we choose “email”



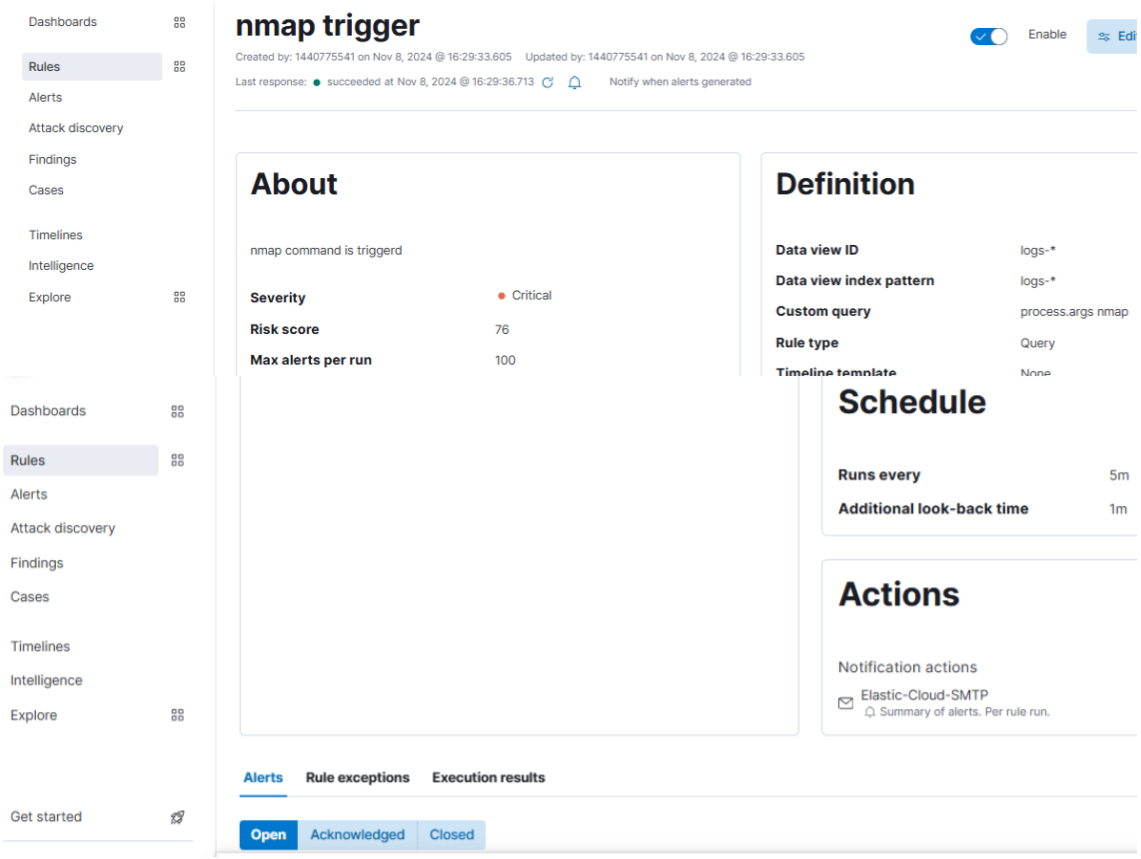
We carry out the configuration and enter an email and a subject



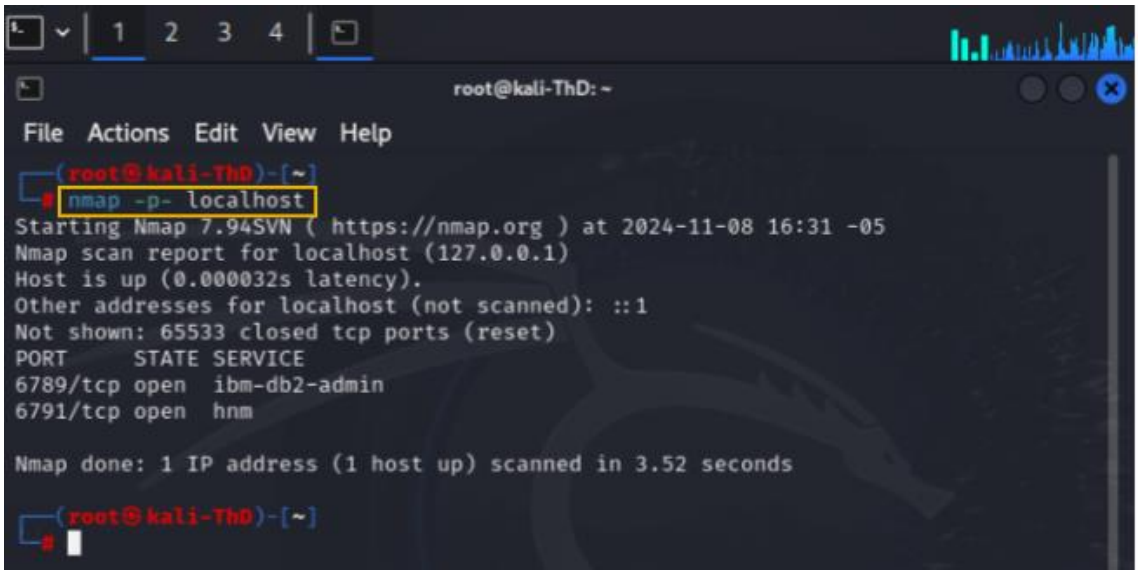
Next, click on "Create & enable rule." You could also create a rule without having it enabled.



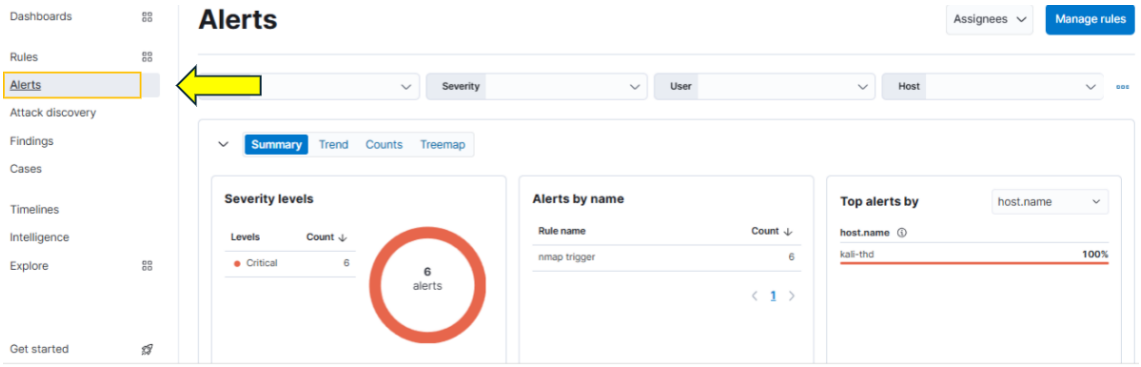
In this way we see our rule created successfully



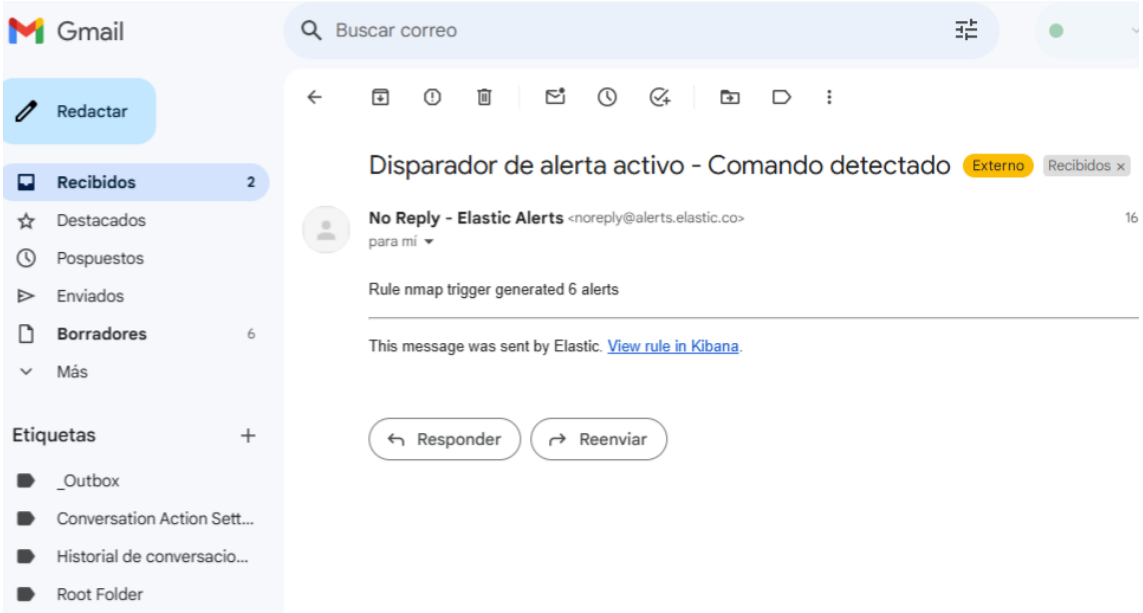
Now we proceed to test that it is working correctly, for this, on our Kali machine we do a scan to *local host* with *Nmap*

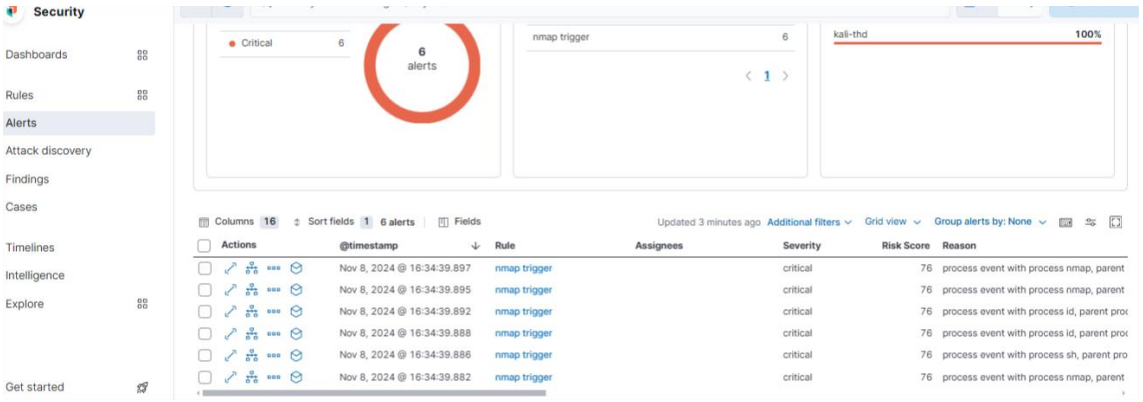


To verify, in the window *elastic* we click on the side menu “Alerts”



And we find the alerts that have been detected by the rule on our Linux machine

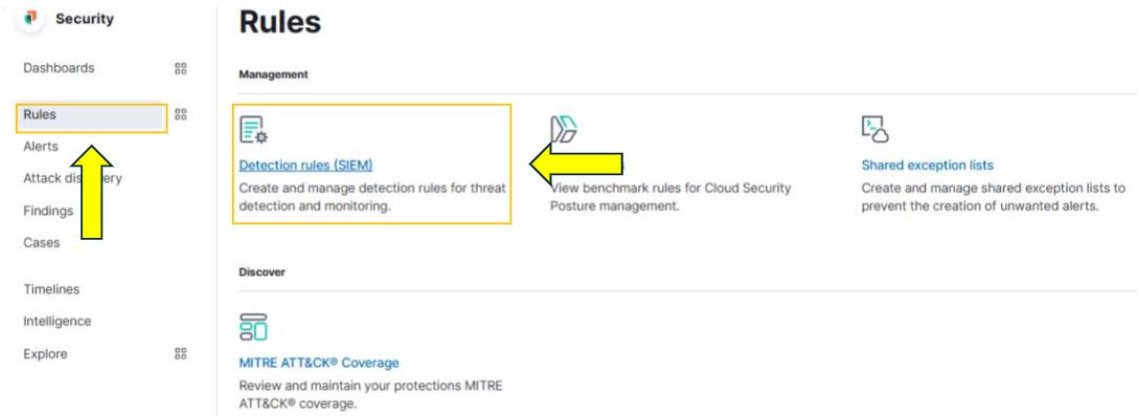




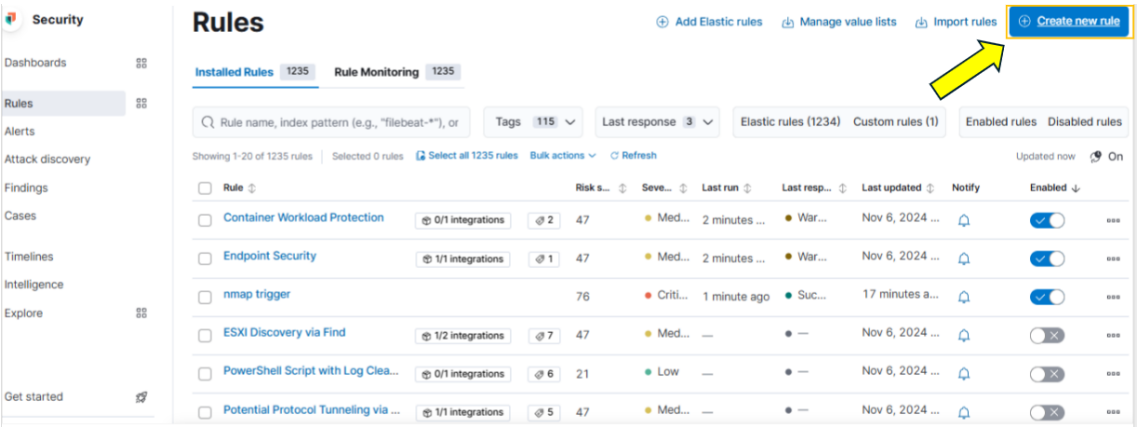
Additionally, we received an email notification of the alert via the trigger.

Now we will create another rule for the command*ping*, following the same procedure shown above.

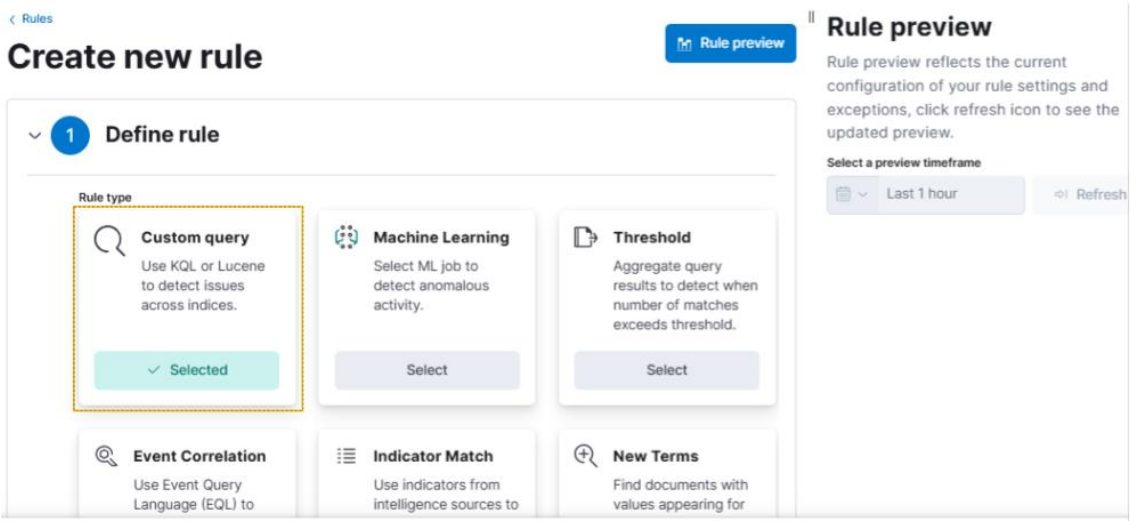
We return to our “Rules” menu on the left menu and select “Detection rules”



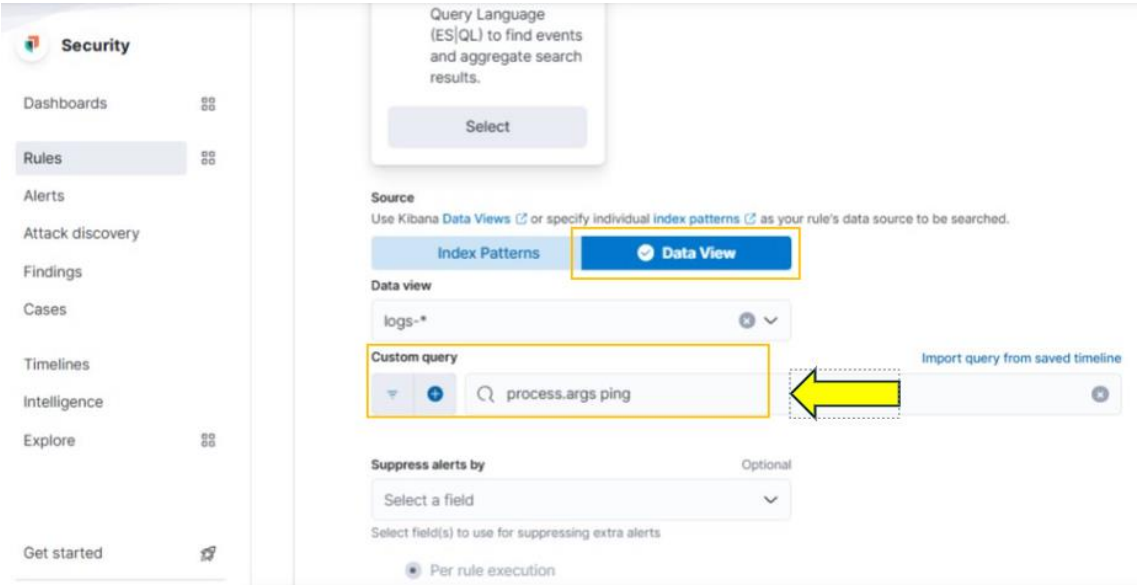
We click on “Create new rule”



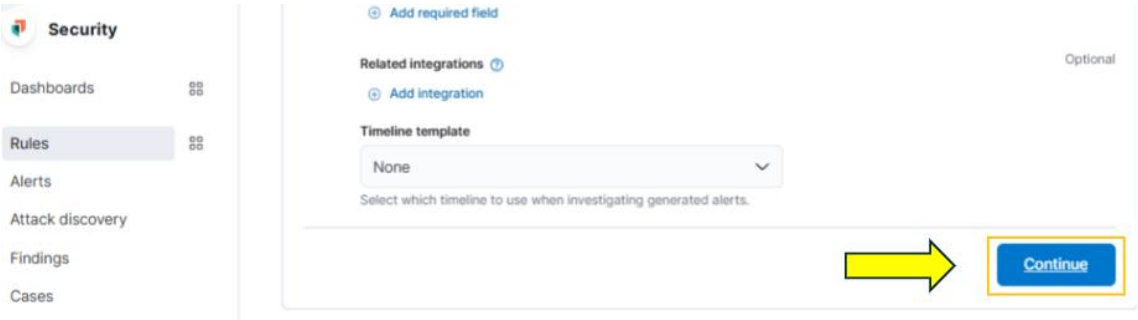
We select “Custom query”



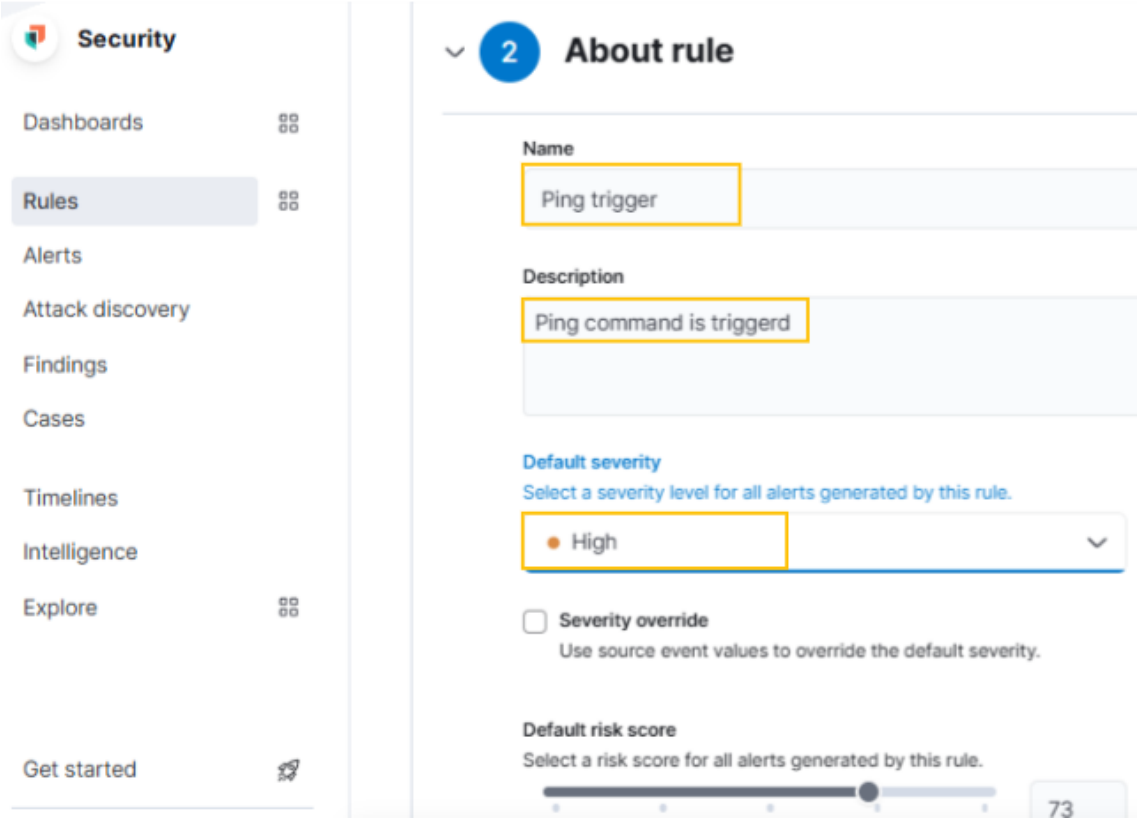
We select the “Data view” tab and add “process.args ping” as a “query”



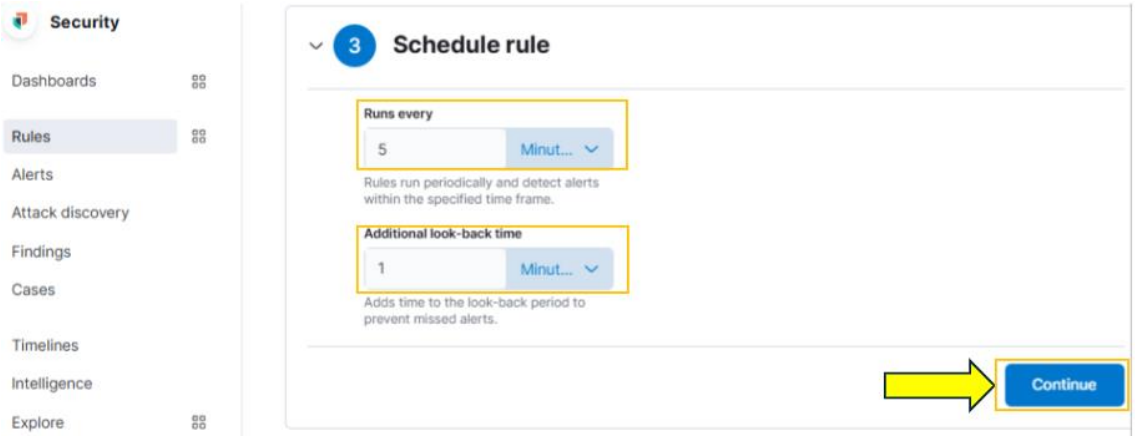
We click on “Continue”



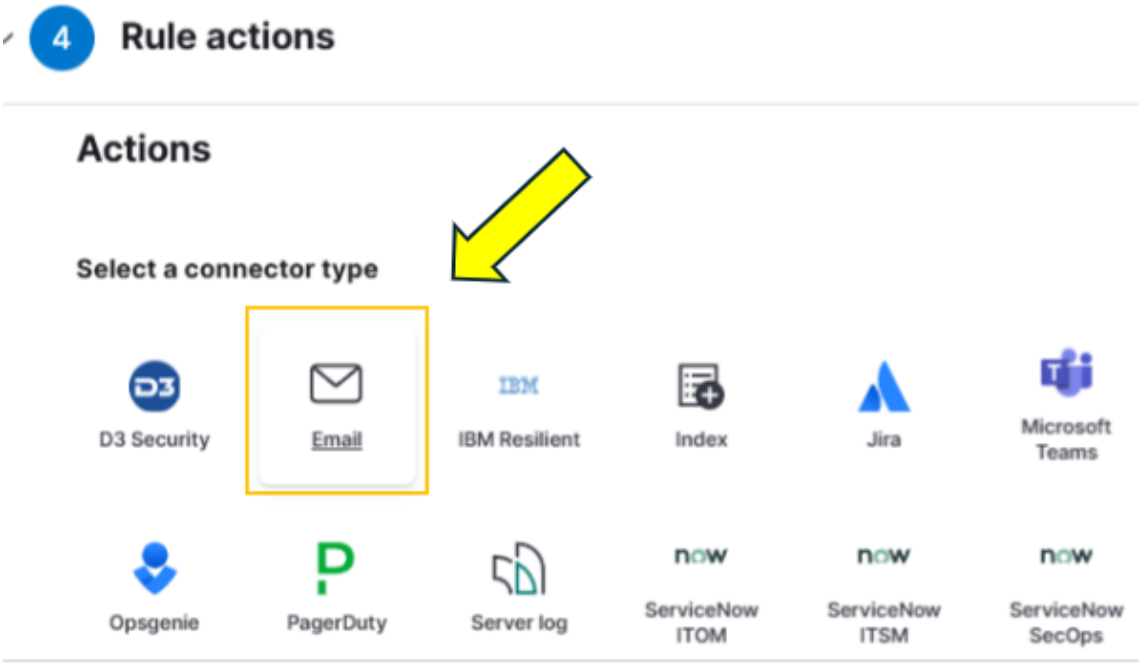
We enter the configuration data



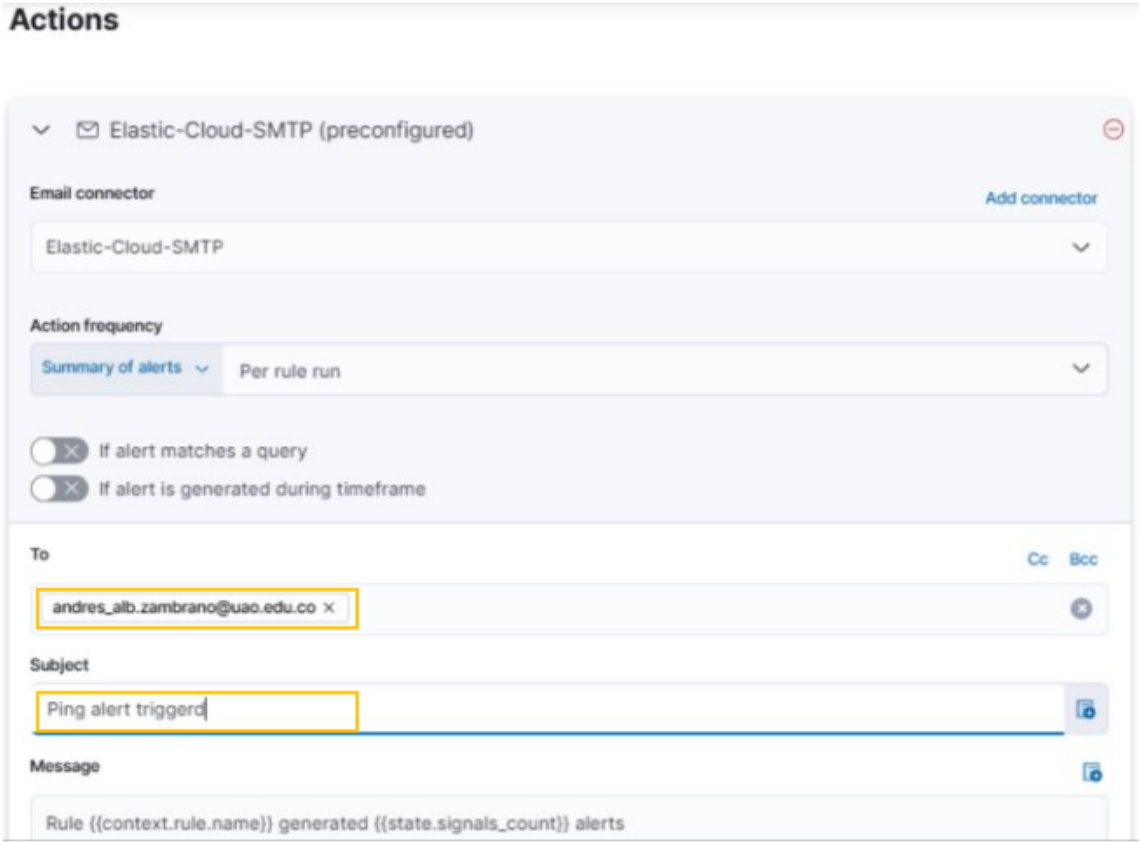
We define the times



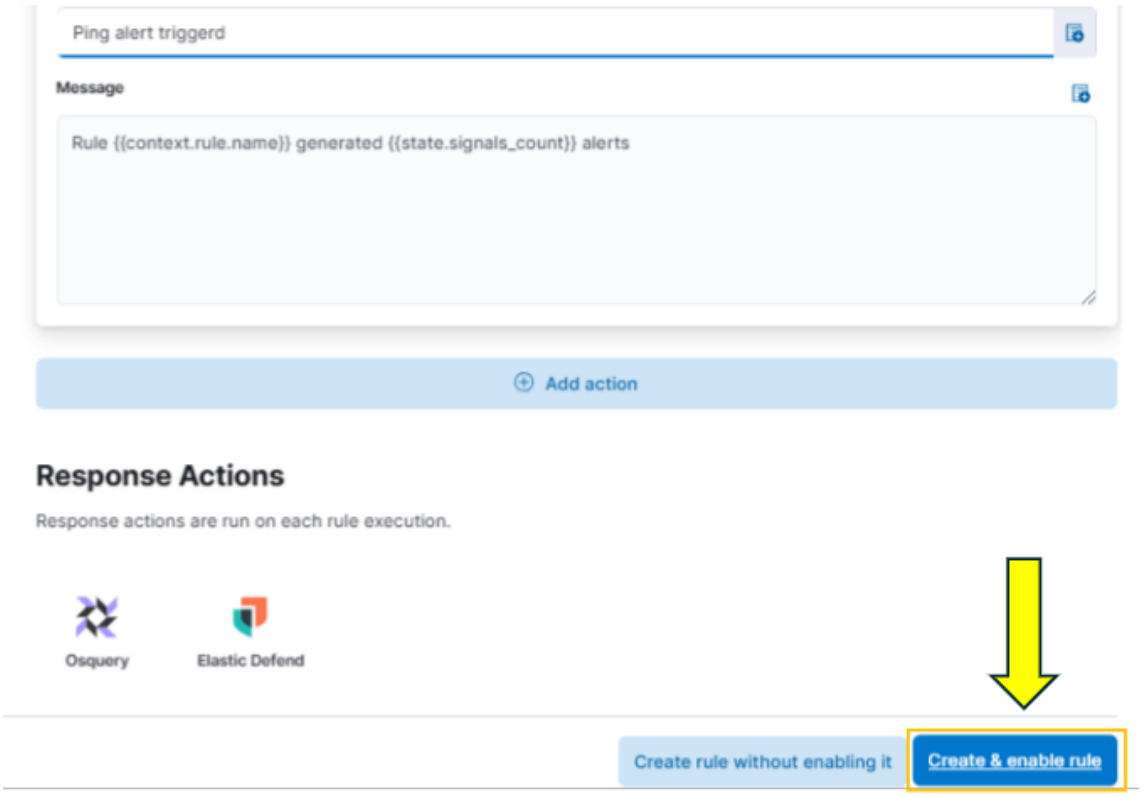
And we choose *E-mail* to receive the notification



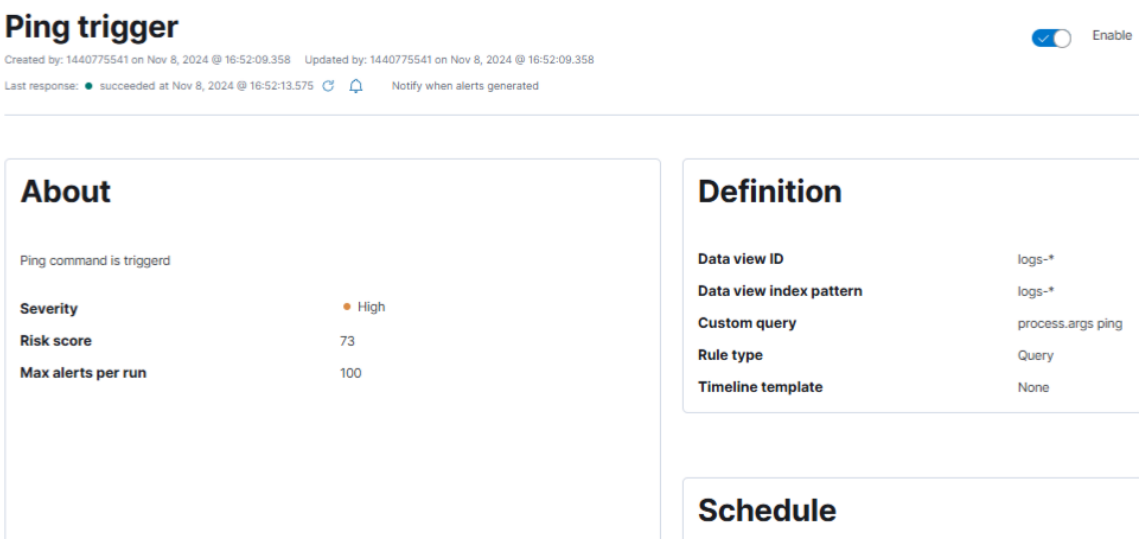
We configure the email and the subject of the message



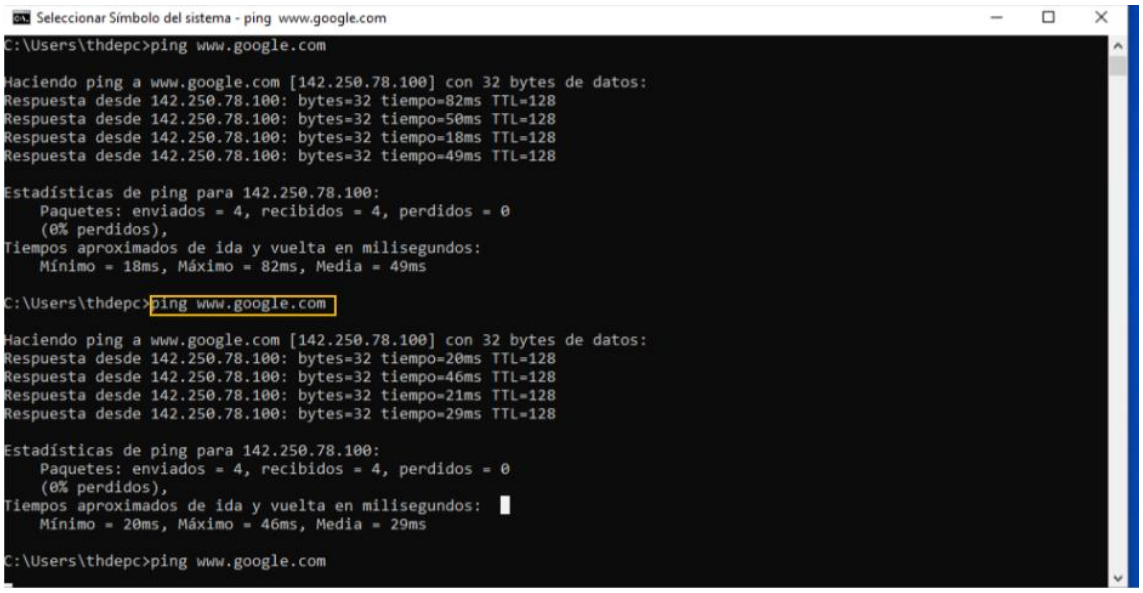
And we click on “Create & enable rule”



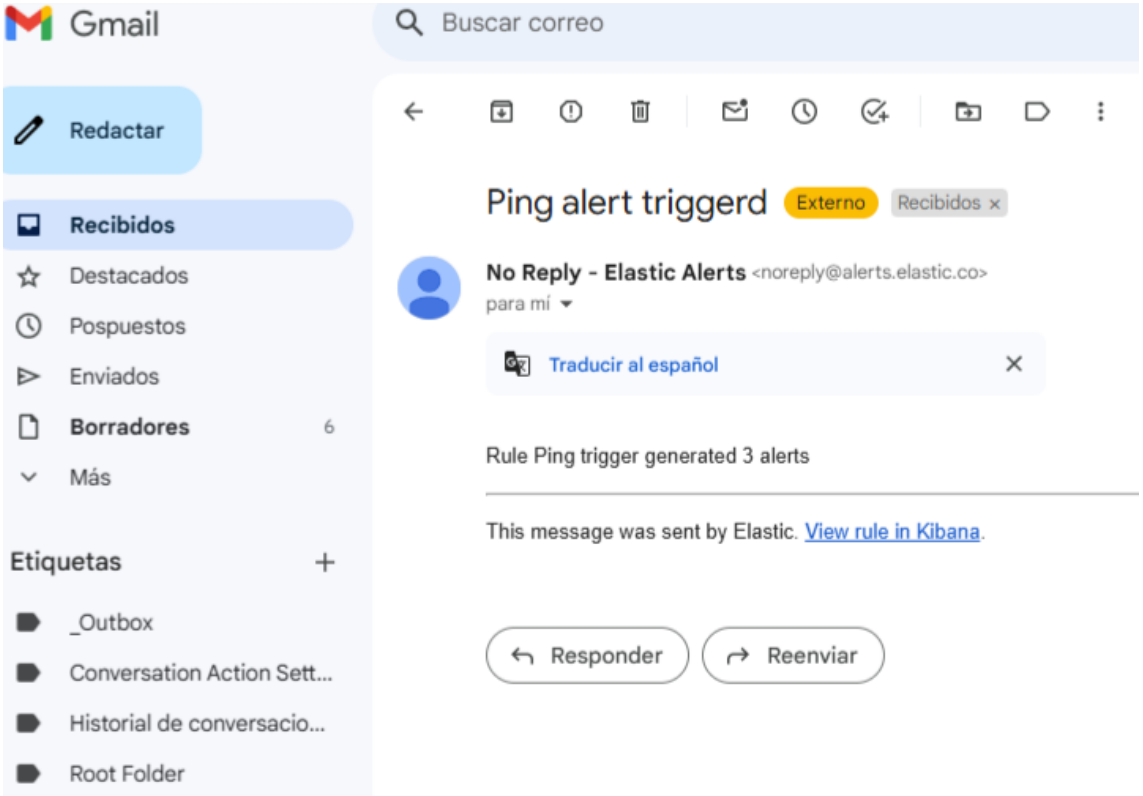
In this way we have our rule created



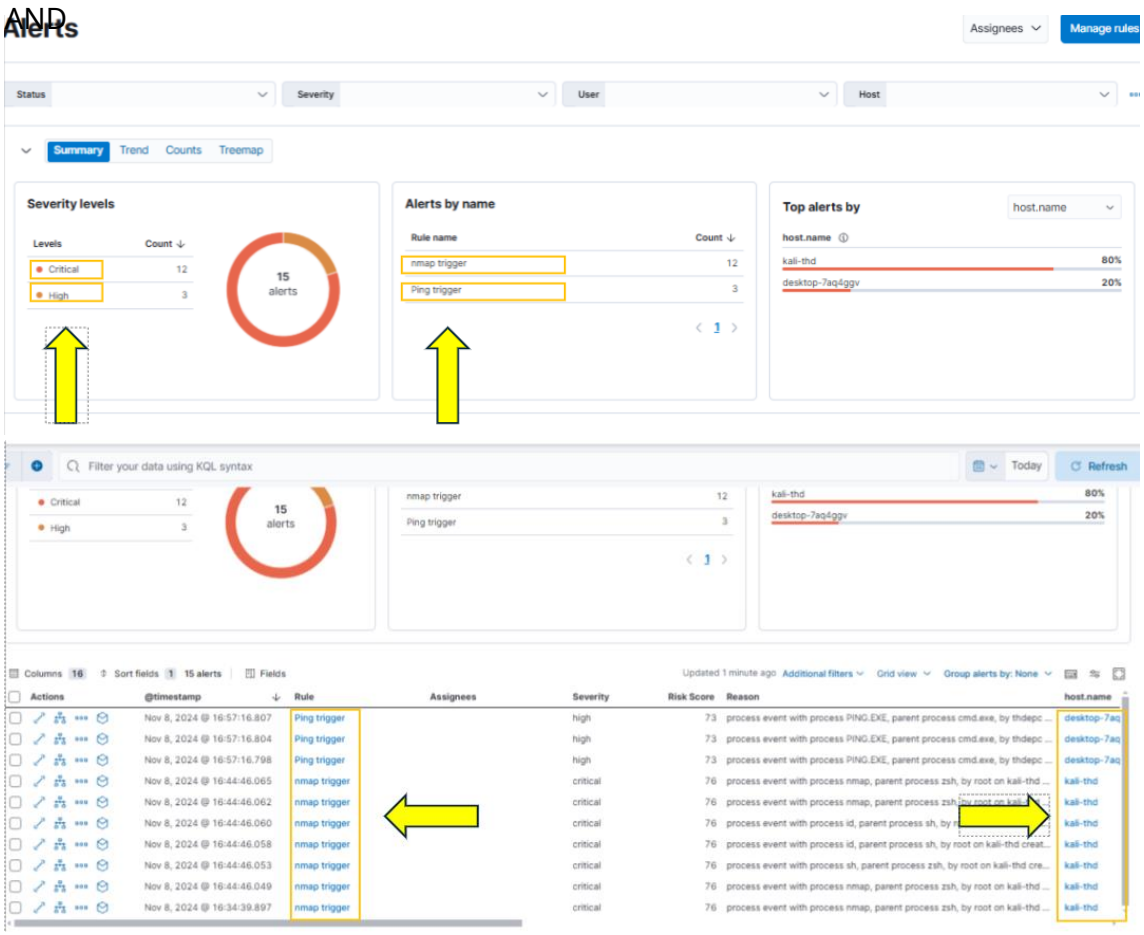
To test our rule we run a *ping* on the Kali and Windows machine



And we received the email with the alert notification



AND



We can see in the right panel the *host name* of Kali and Windows 10 machines and the list of events presented.