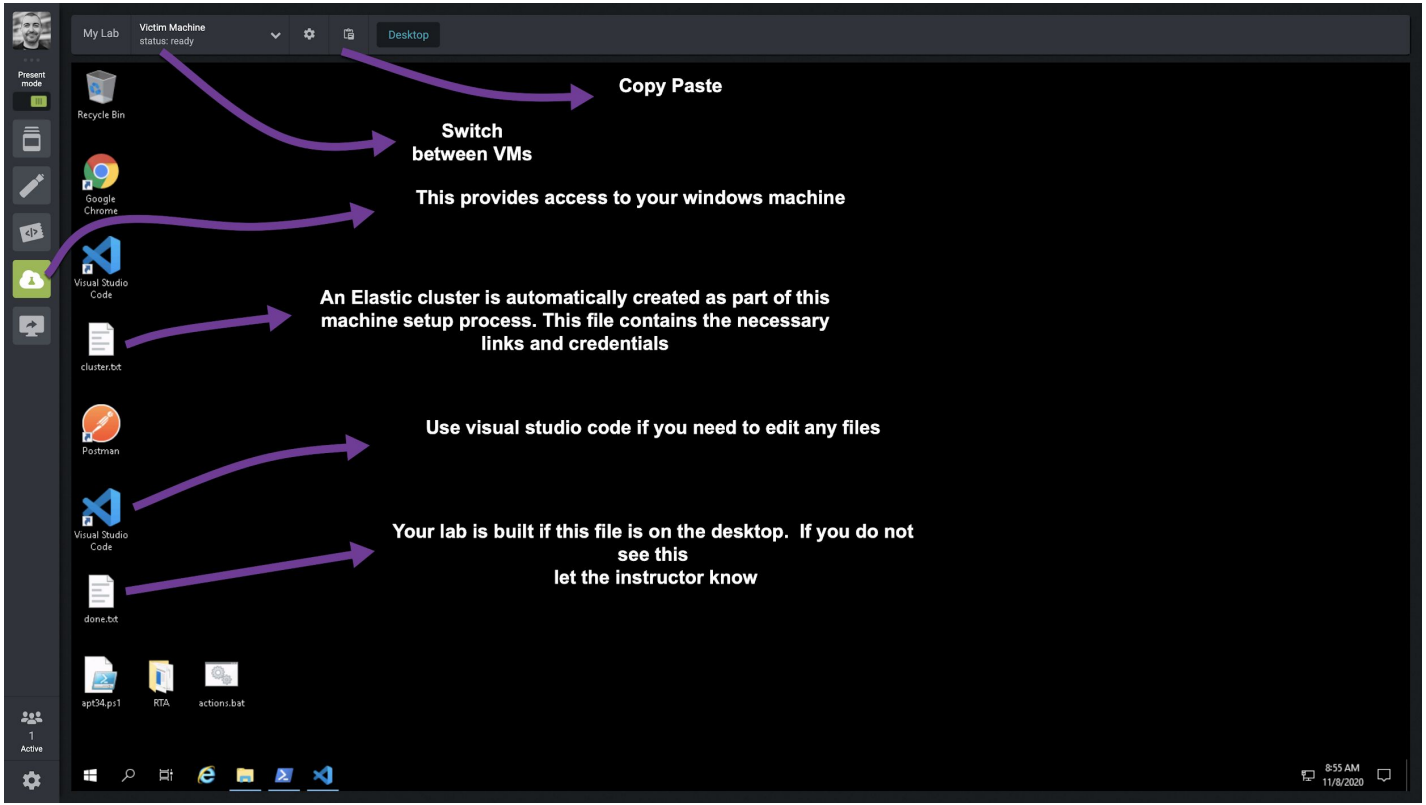
Adversary Emulation Labs

Lab 1 - Data Collection

10 - 15 Minutes
Overview and Preparation

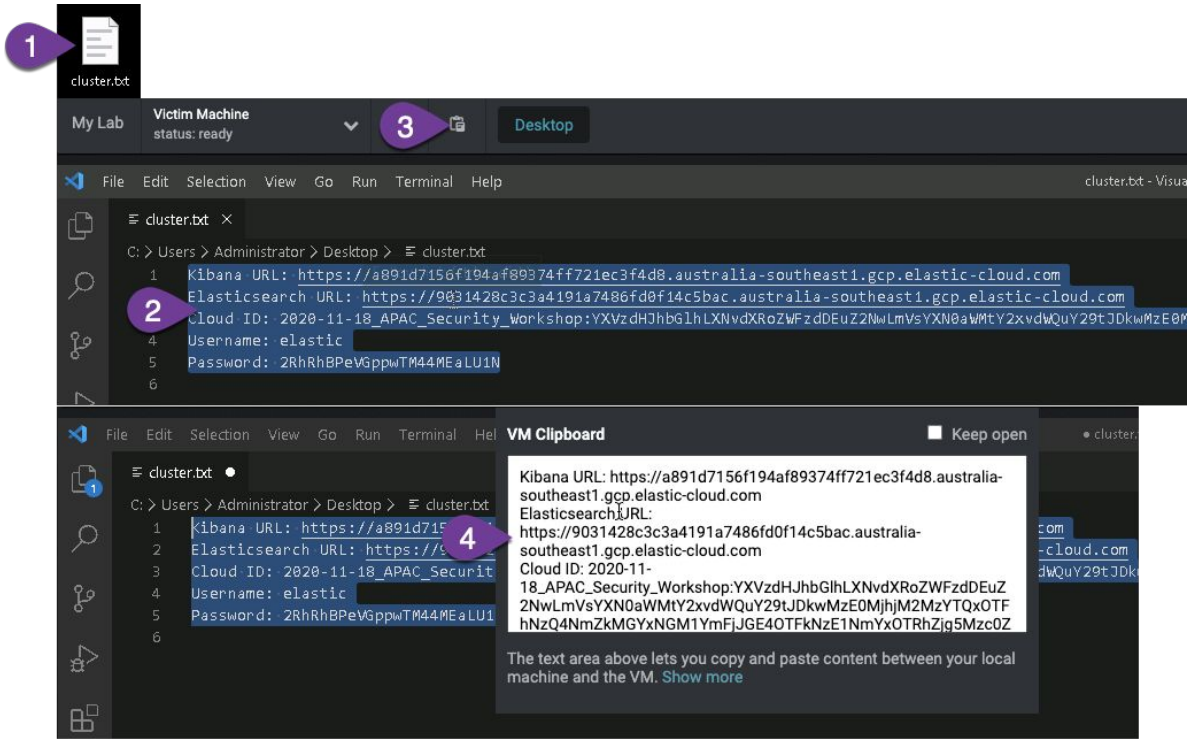
Strigo Overview



Login to Kibana

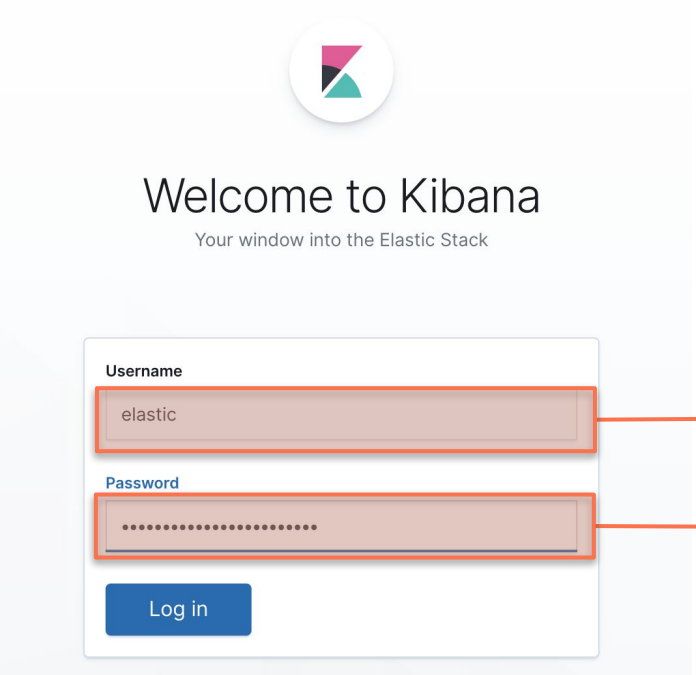
Open **cluster.txt** on your desktop and copy the credentials

On a Mac use “Control + C” not “Command + C”



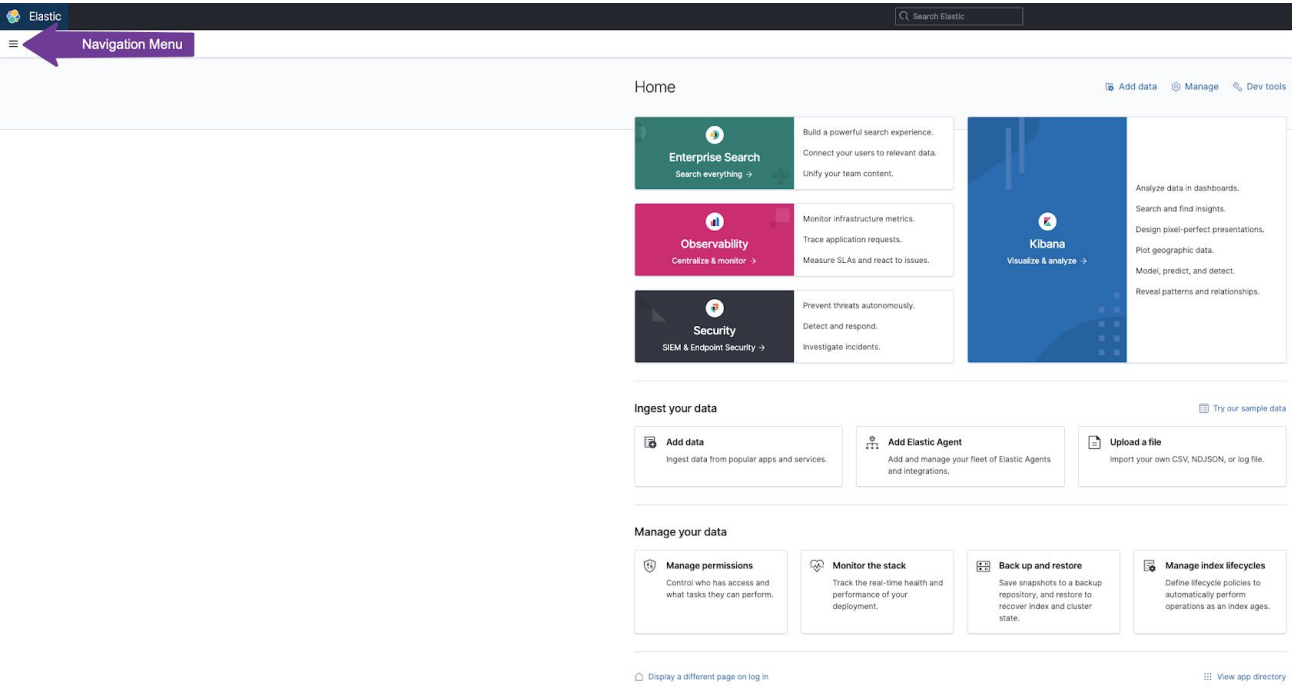
Open browser on your machine and log into Kibana

As user ‘elastic’



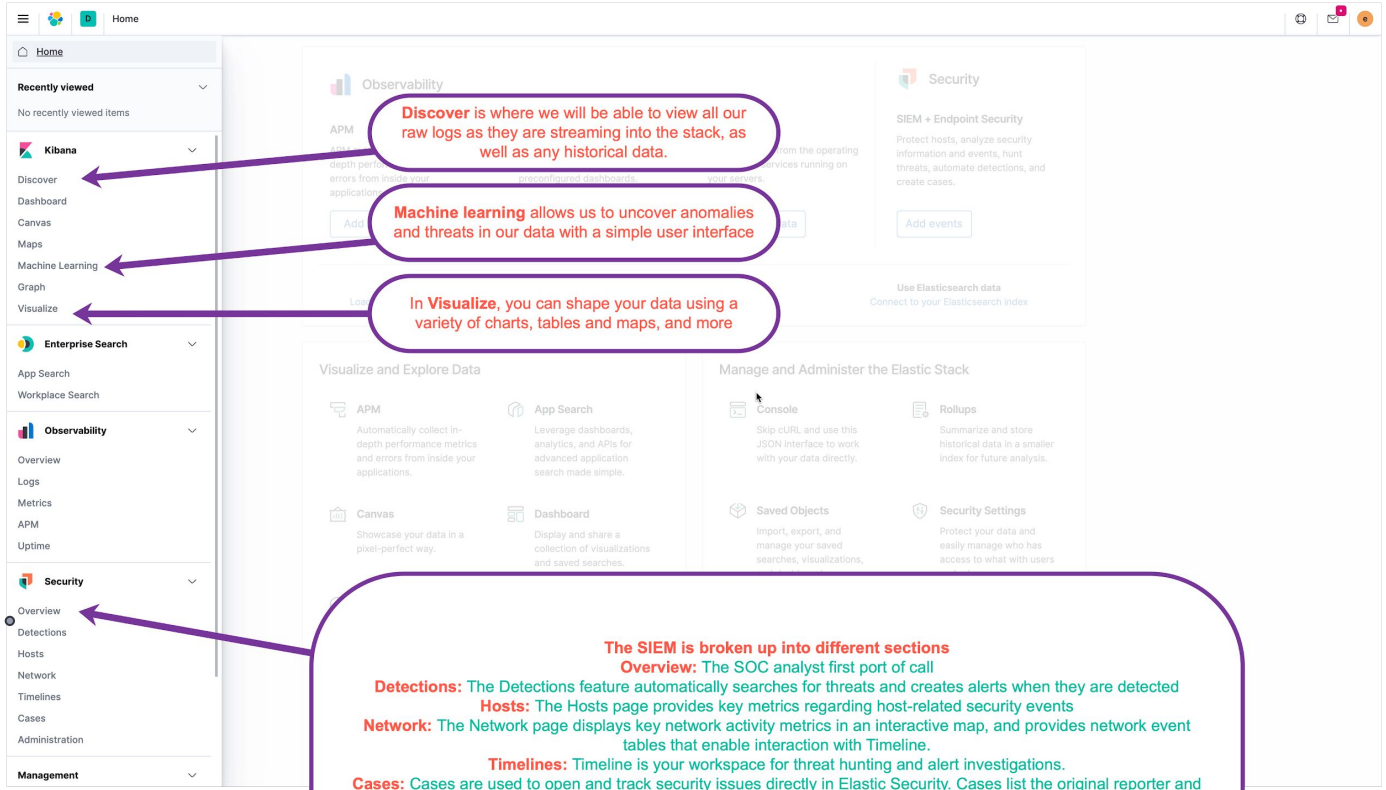
1. Username is: elastic
2. Copy-n-Paste the Cloud elastic user password copied from slide before

Open Navigation Menu



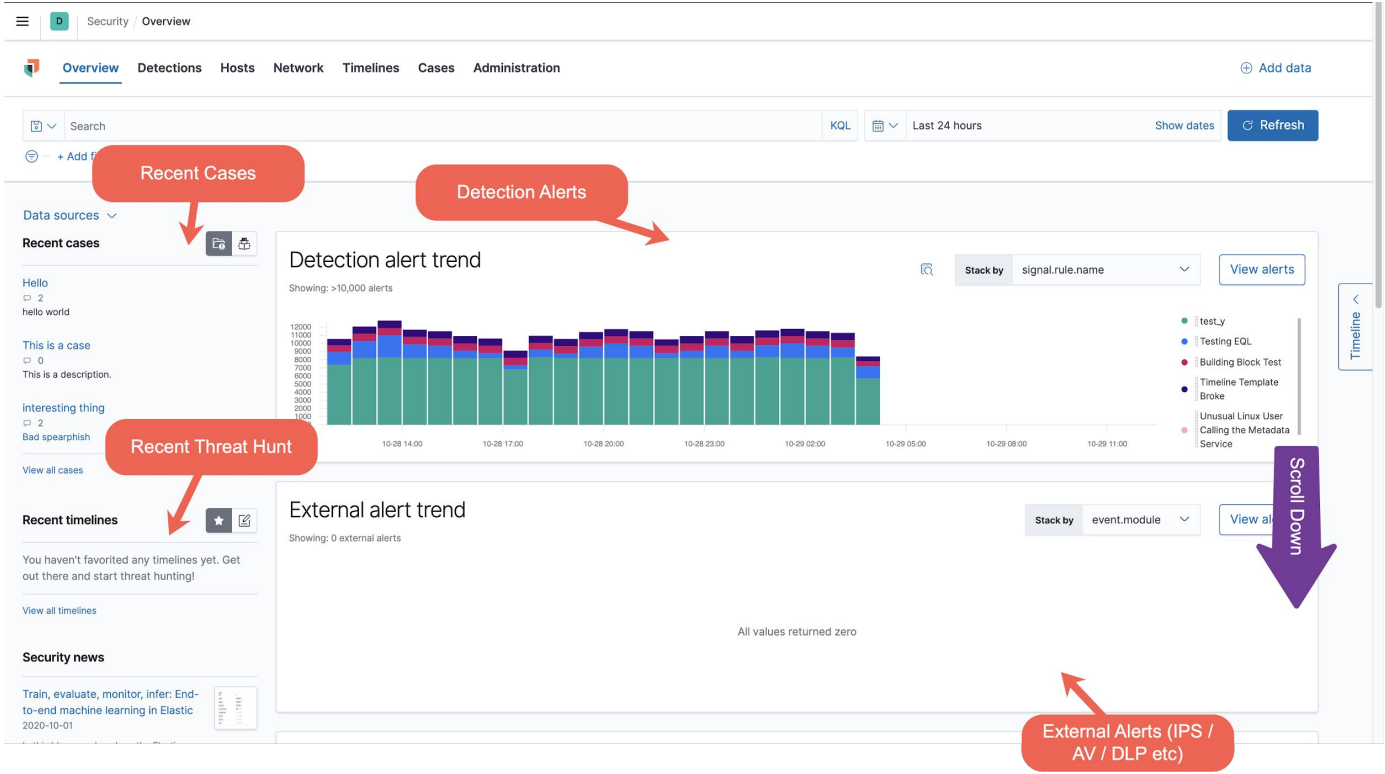
Kibana Menu

Click Overview

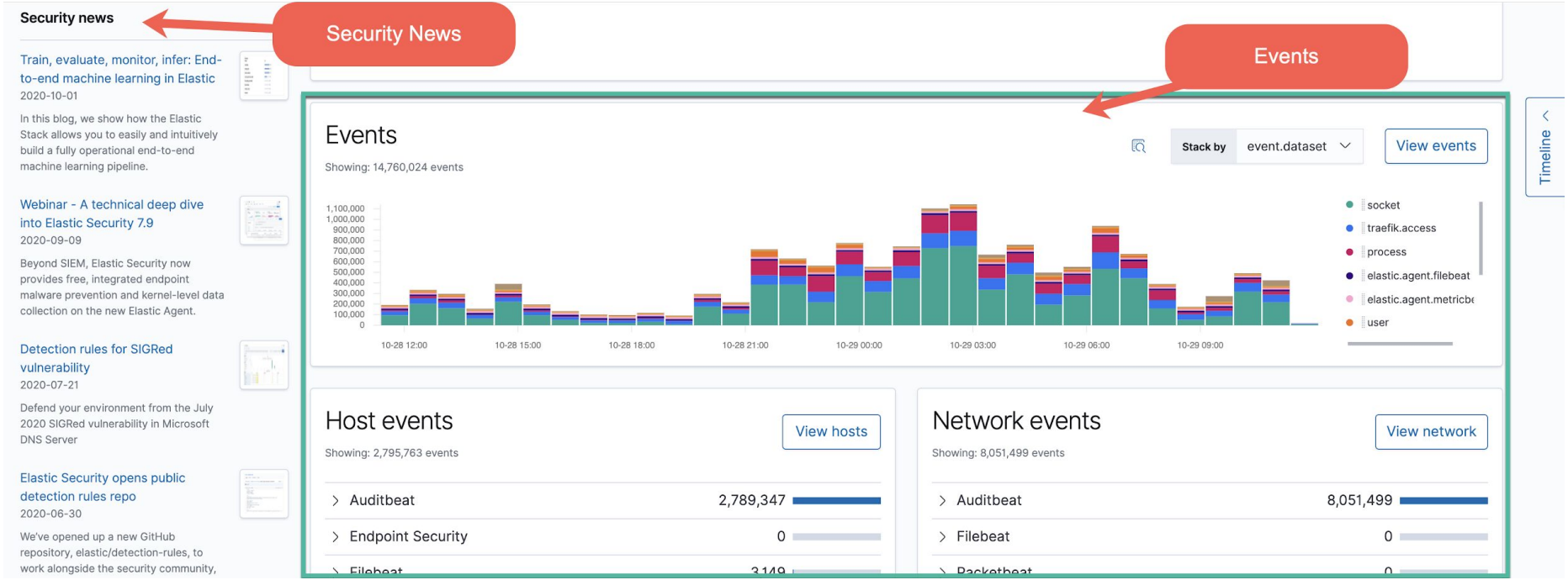


elastic

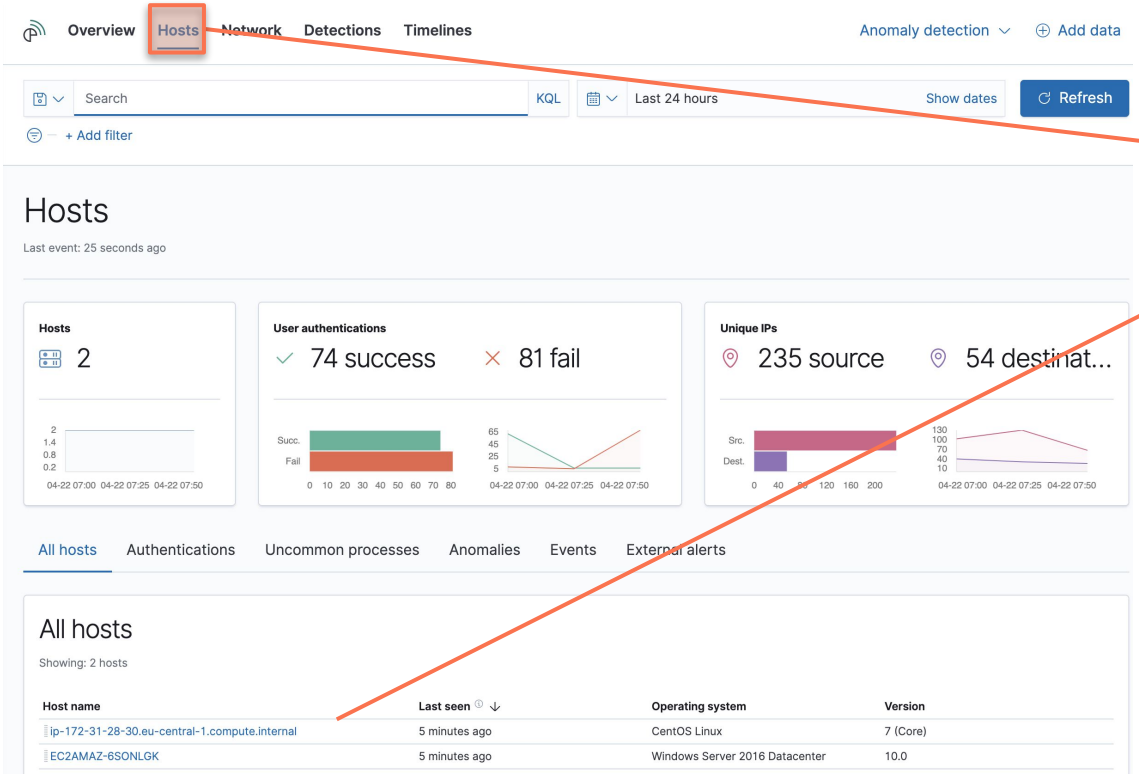
SIEM Overview



SIEM Overview



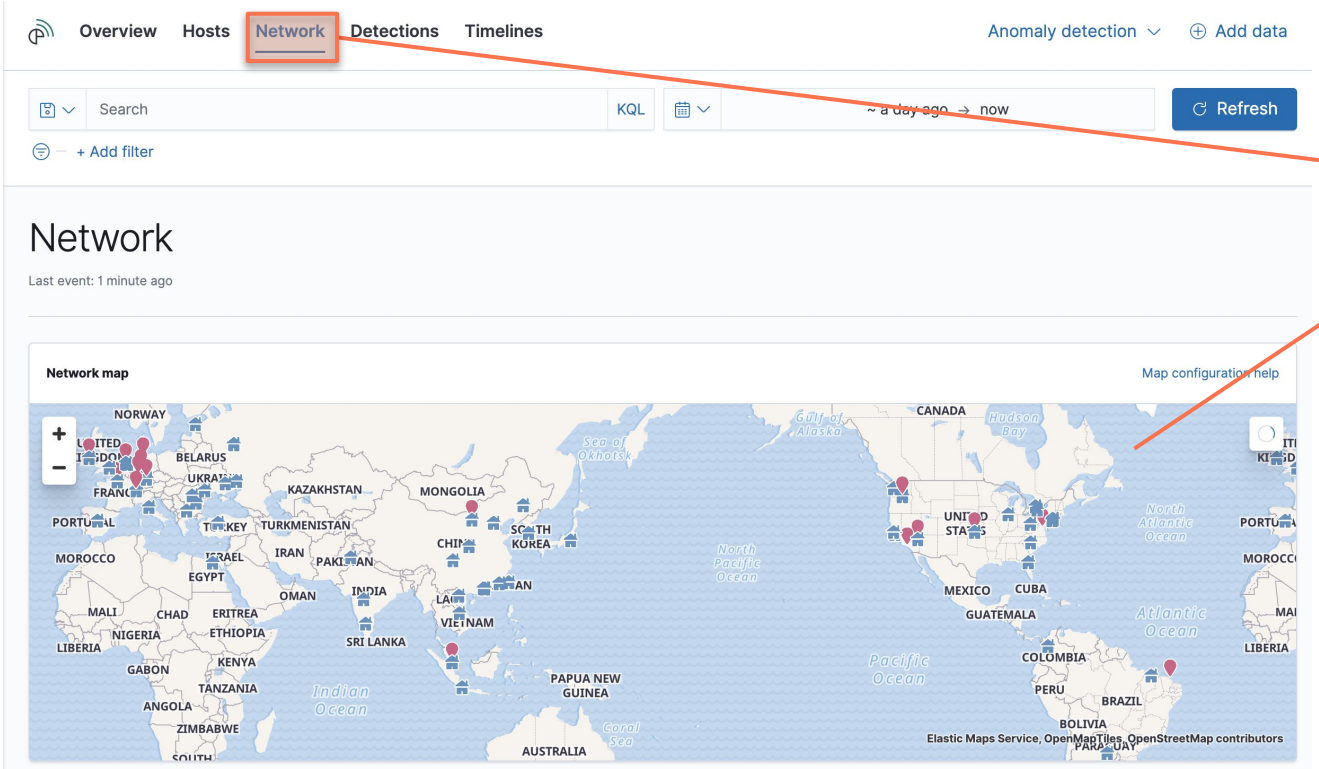
Hosts Overview



1 Click on the Hosts Tab

2 Notice you have VM's available to interrogate. Click on any of the hosts to filter the view on the specific host.

Network Overview



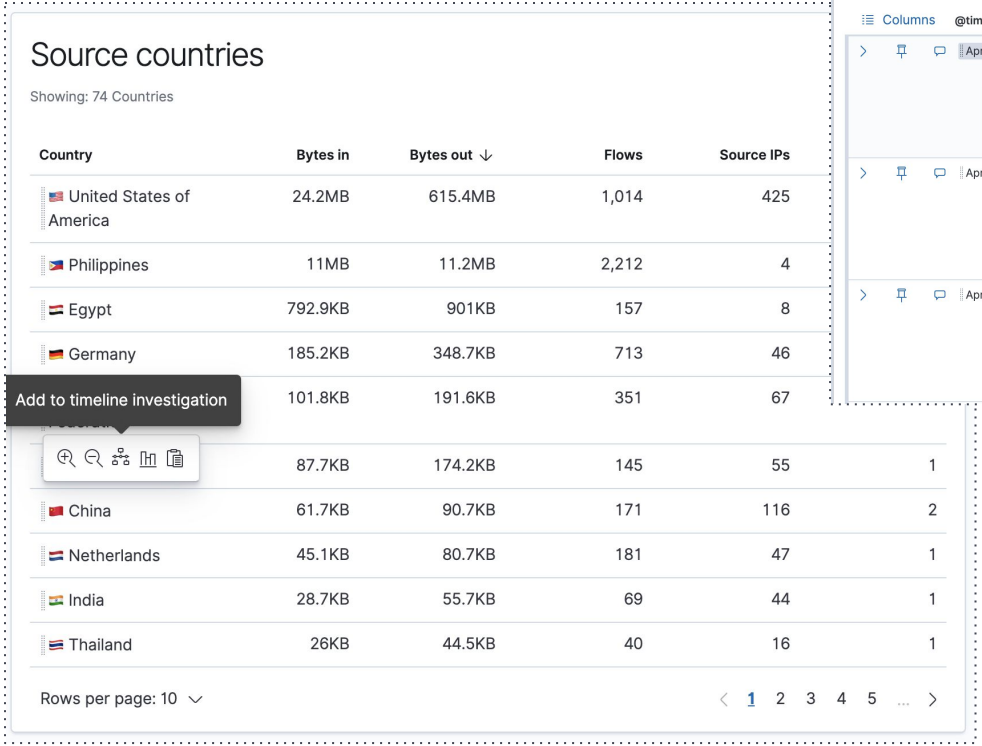
1. Click on the Network Tab

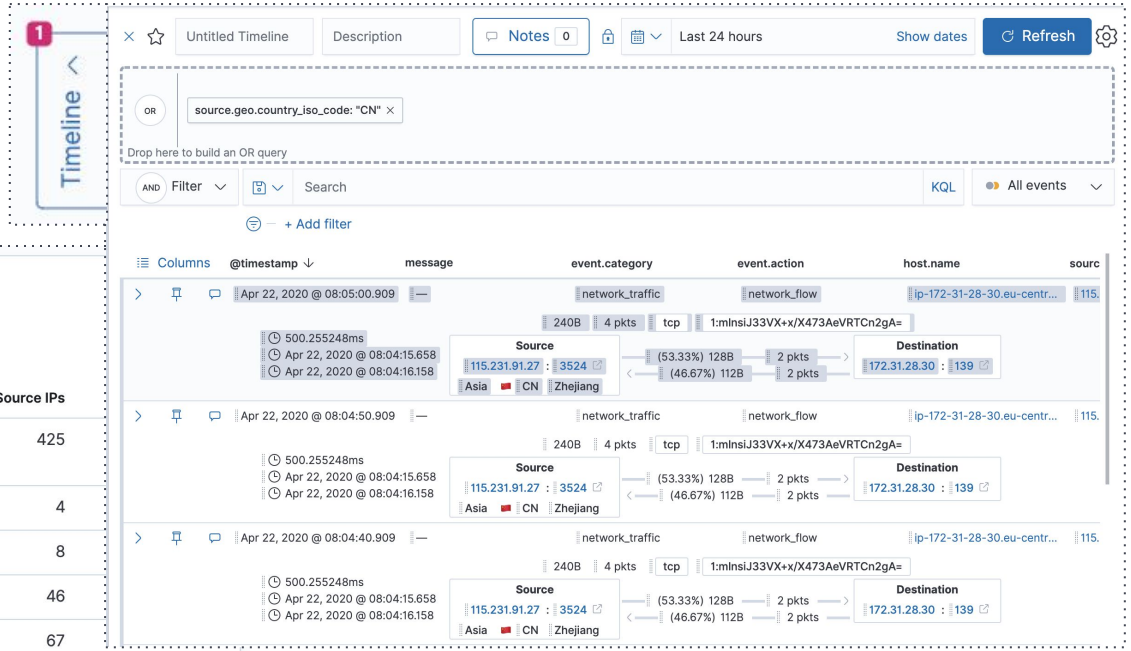
2. Notice the rich network information displayed thanks to geo-IP enrichment at ingest

elastic

Timeline

Create an Investigation

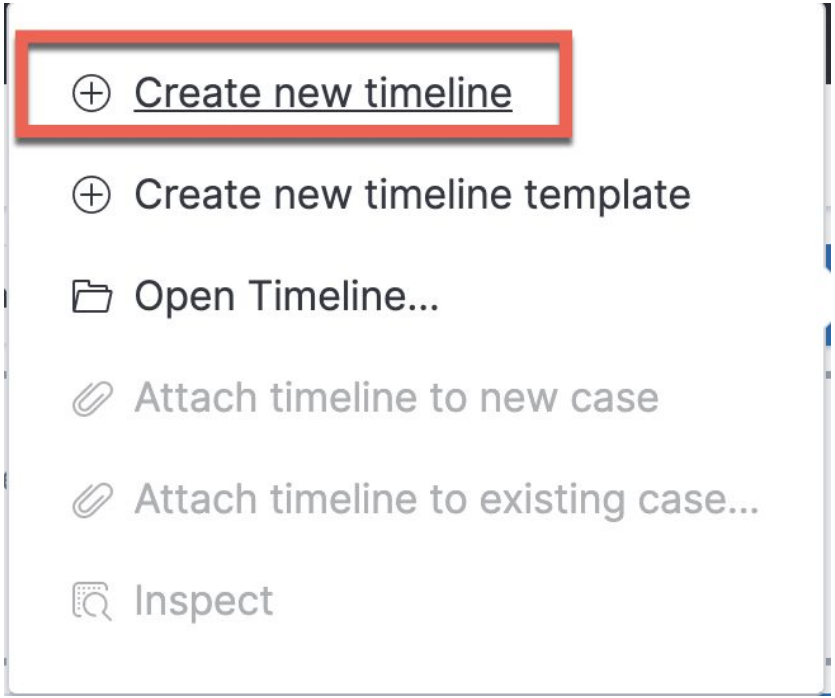




Hover over any field (e.g, Source Country United States of America) then select add to timeline investigation and click the Timeline button

Timeline

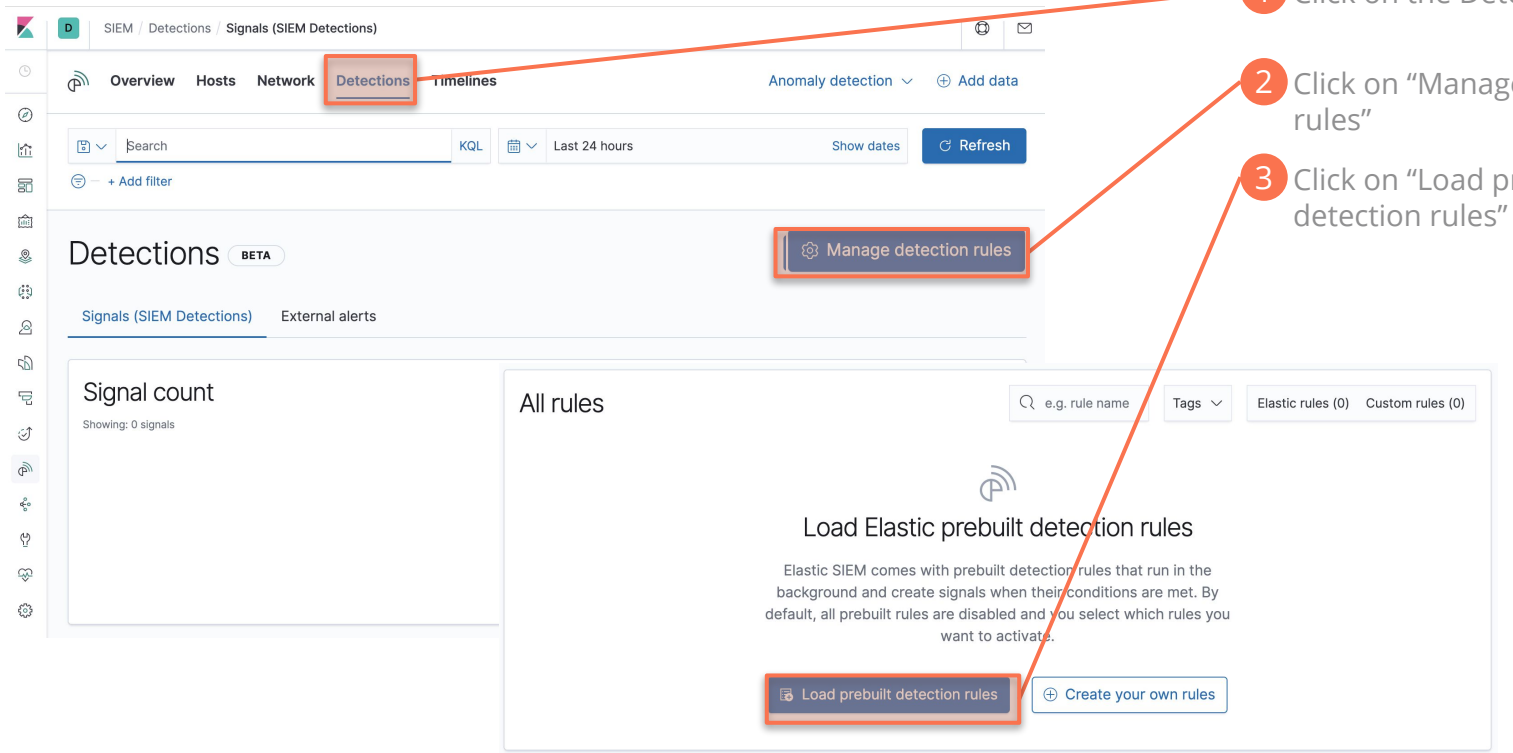
Create new timeline to clear view



elastic

Detections

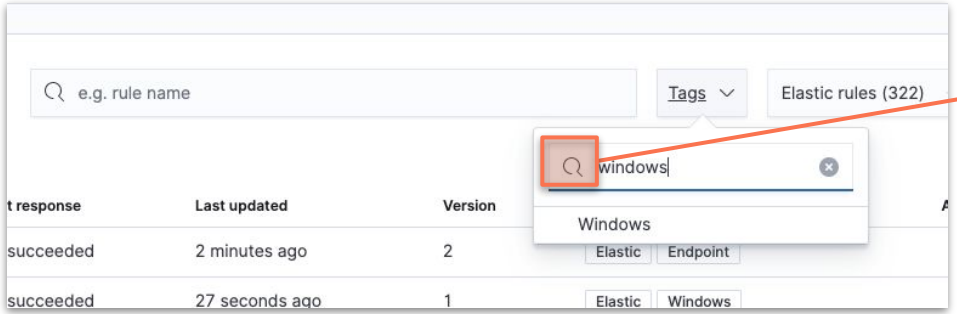
Enable OOTB Detections



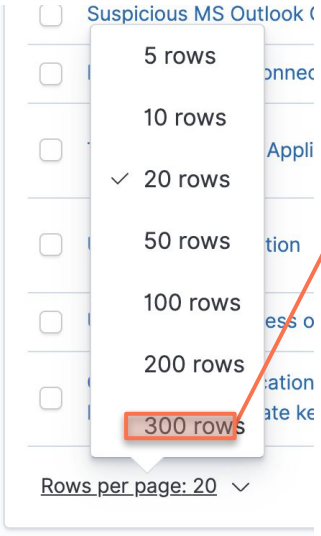
1. Click on the Detections Tab
2. Click on “Manage detection rules”
3. Click on “Load pre-build detection rules”

Detection

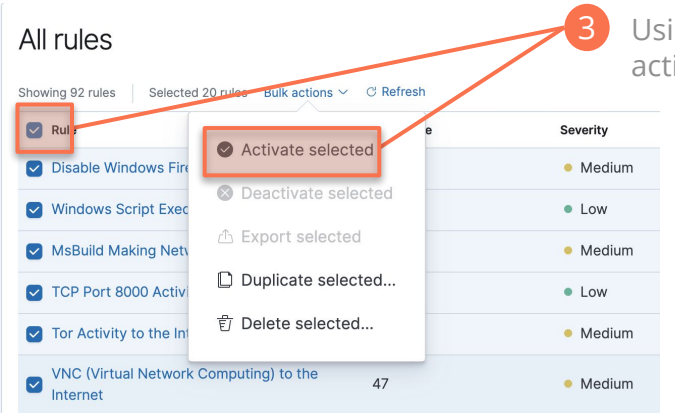
Enable OOTB Detection Rules for **Windows** & **Network**



1 After spending some time going through the OOTB rules, click the tags **Windows**, **Network**
Note: "Tags" is an implicit "AND" not "OR", you will need to unselect each tag



2 Select "300 rows"

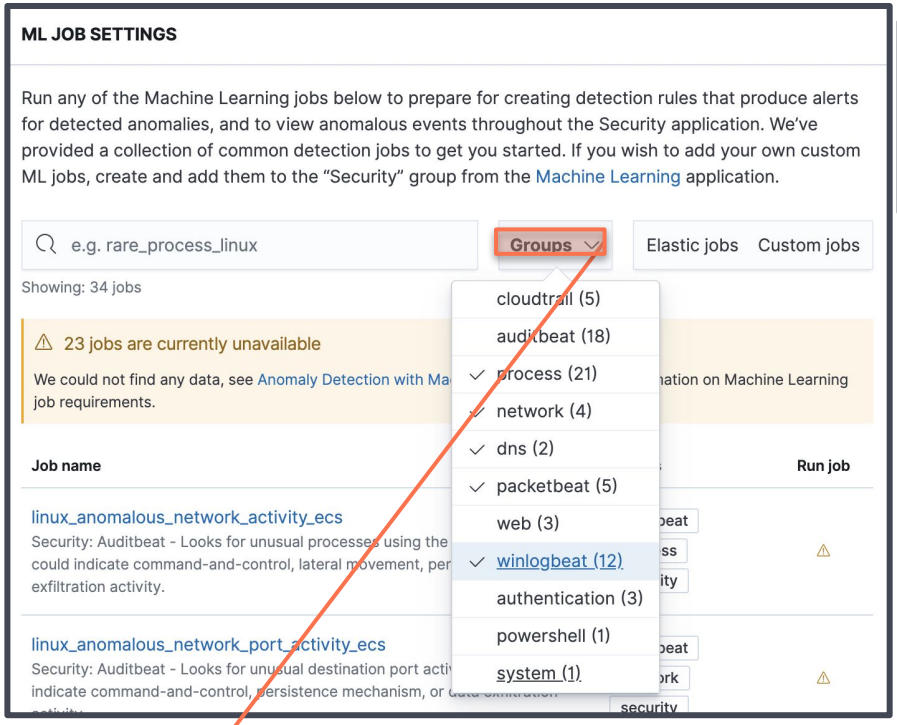


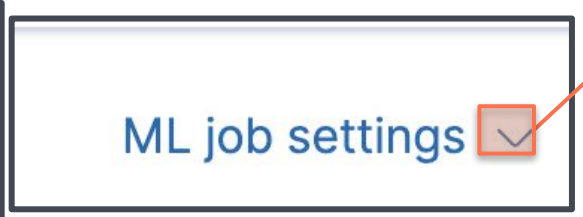
3 Using "Bulk Actions" activate all rules

elastic

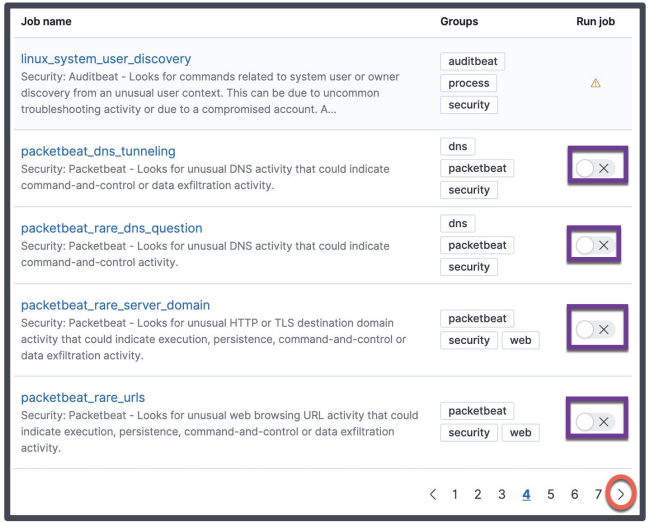
Detection
Enable OOTB Machine Learning Jobs

OPTIONAL





1 Select drop down of ML Job Settings



4 Enable all of the rules

2 Select **Process, Network, DNS, Packetbeat and winlogbeat**

elastic

Run RTAs on the Windows VM

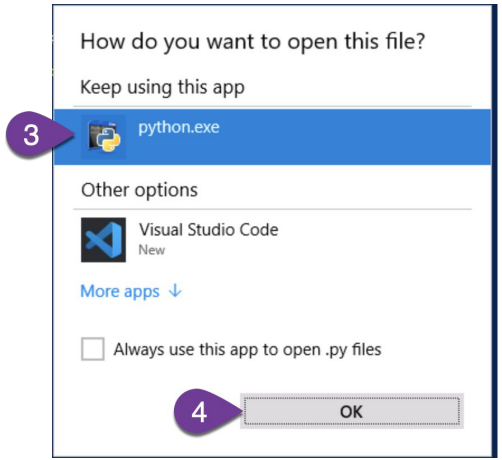
To generate Signals

OPTIONAL

Administrator: Windows PowerShell

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> cd C:\Users\Administrator\Desktop\RTA\
PS C:\Users\Administrator\Desktop\RTA> cd .\red_ttp\
PS C:\Users\Administrator\Desktop\RTA\red_ttp> .\msbuild_network.py
PS C:\Users\Administrator\Desktop\RTA\red_ttp> .\disable_windows_fw.py
PS C:\Users\Administrator\Desktop\RTA\red_ttp>
```



1. On your Windows VM in Strigo, execute two of the RTA scripts: msbuild_network.py and disable_windows_fw.pw

2. Within a few minutes, notice the Signals generated in the Detections view of the SIEM app

Apr 22, 2020 @ 07:16:18.890	Trusted Developer Application Usage	1	query	low	21	sysmon
Apr 22, 2020 @ 07:15:37.021	MsBuild Making Network Connections	1	query	medium	47	sysmon
Apr 22, 2020 @ 07:13:24.567	Disable Windows Firewall Rules via Netsh	1	query	medium	47	sysmon

elastic

OPTIONAL

Search Basics - KQL

KQL

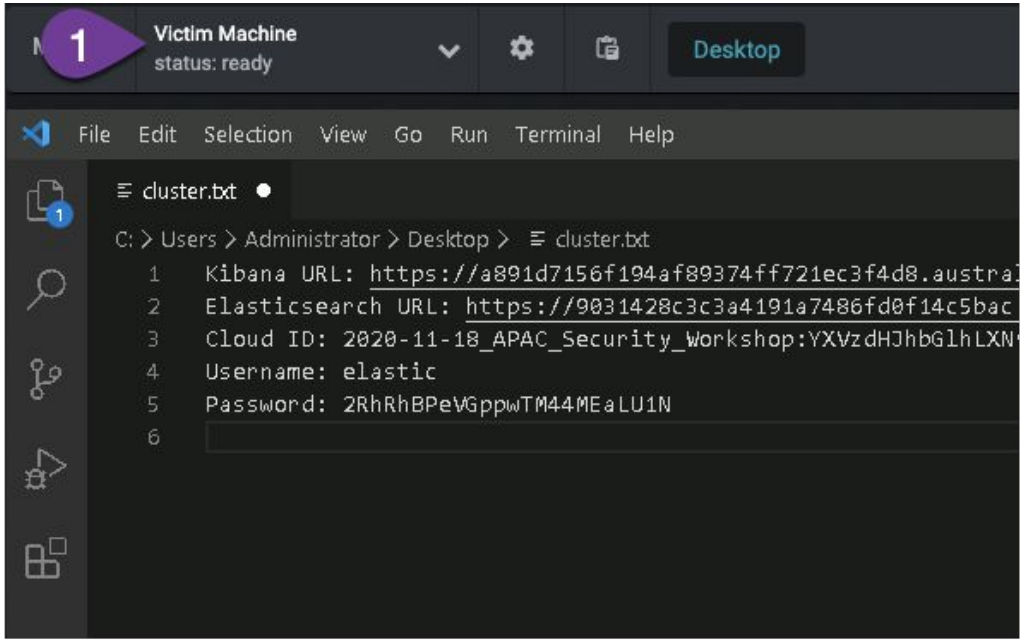
```
error - Free Text Search
response: 200 - Field Search
(error and response: 200) - Boolean AND / OR
response:200 and not (extension:php or extension:css) - NOT with OR
response:>=300 and response:<=500  - Range query
agent.keyword : *MSIE* - Wildcard Search
machine.os*:windows 10  - Wildcard field match
```

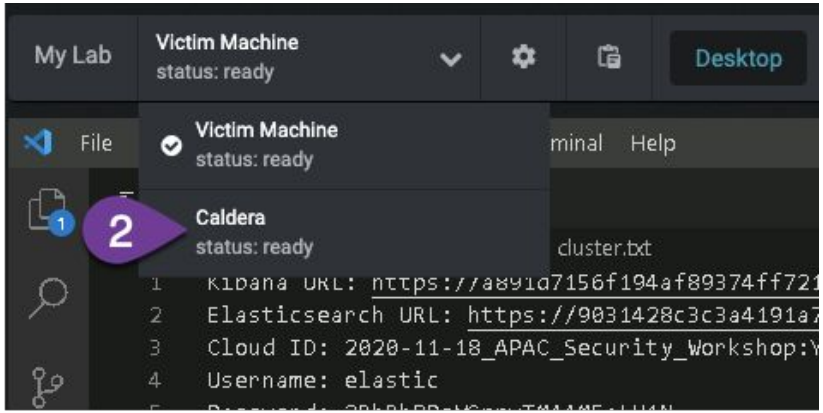
End Lab 1

Lab 2 - Implant Deployment

10 - 15 Minutes
Initial Implant and visibility

Switch to Caldera

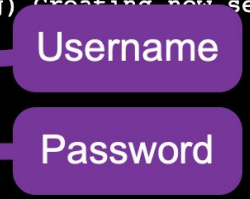




Retrieve Login credentials for caldera

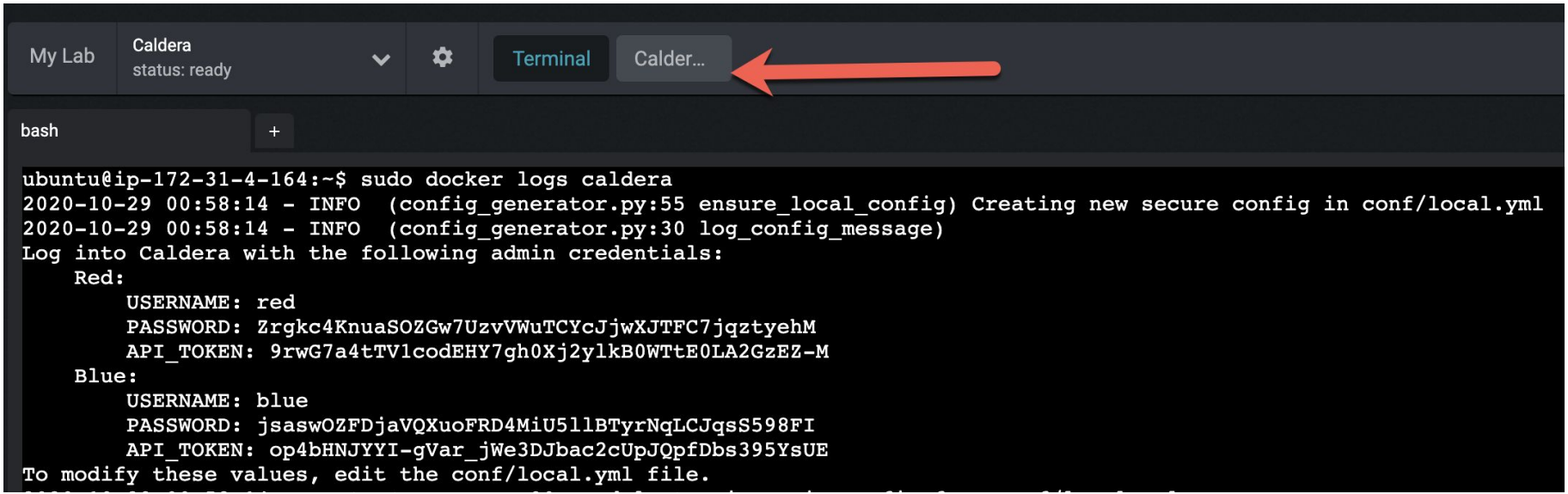
Enter command: **sudo docker logs caldera**
Copy credentials

```
ubuntu@ip-172-31-0-202:~$ sudo docker restart caldera
caldera
ubuntu@ip-172-31-0-202:~$ sudo docker logs caldera
2020-10-01 01:27:01 - INFO  (config_generator.py:55 ensure_local_config) Creating new secure config in conf/local.yml
2020-10-01 01:27:01 - INFO  (config_generator.py:30 log_config_message)
Log into Caldera with the following admin credentials:
    Red:
        USERNAME: red
        PASSWORD: X0f7eoXAZMfjnKxVUEA2PBtmeq8mCdUrz4EGdWGYzzE
        API_TOKEN: 3VgSr2Bx_OWcVrGKufnNhUUbuT1AUpknrqEekr73fpk
    Blue:
        USERNAME: blue
        PASSWORD: o7nLQmfT_1CecjsXej88vOKBWgHeHMaZh0WK7uWyOkI
        API_TOKEN: JQ4ZhwWxBLGxFohlgIVdNQiZswv21_HVo5MvnQ9Pfgk
```



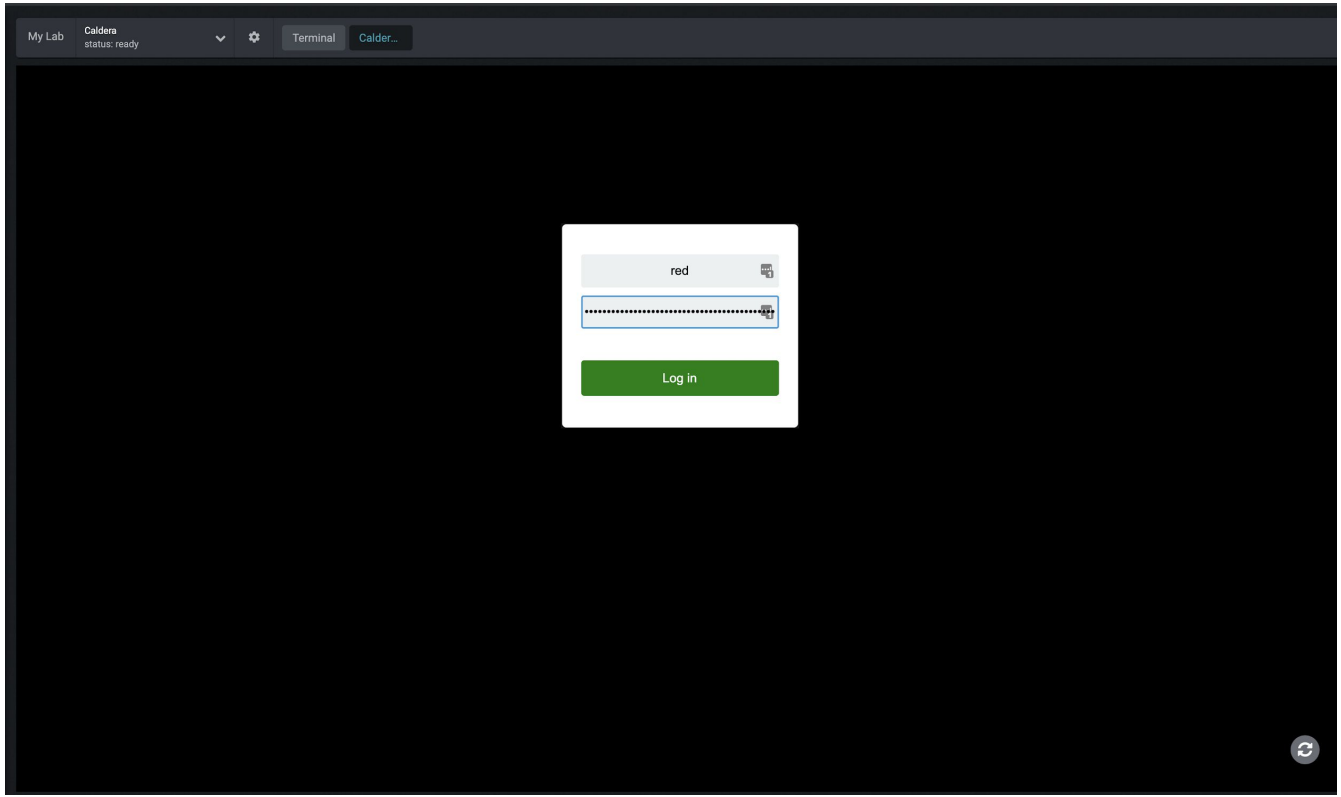
elastic

Open Caldera Web Console



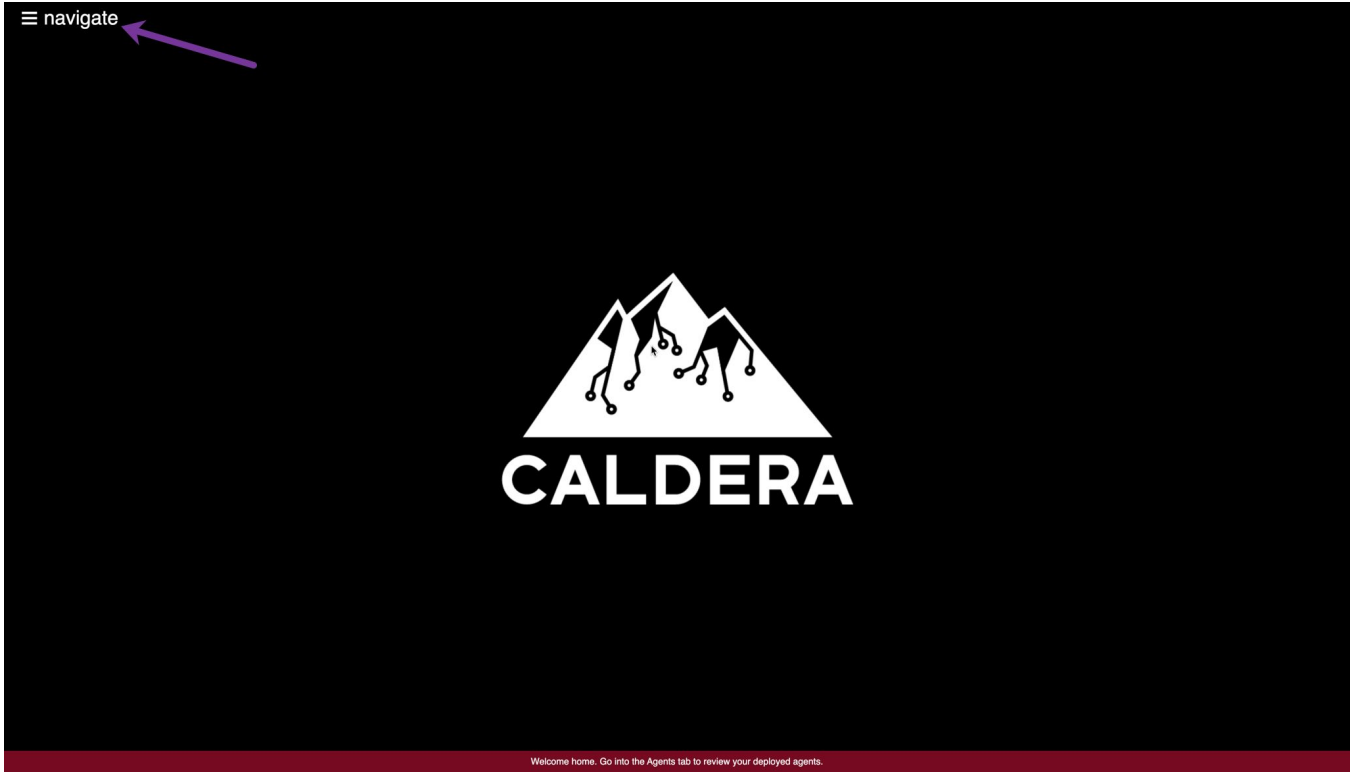
Login

Username **red** password copied from the console



elastic

Caldera Navigation

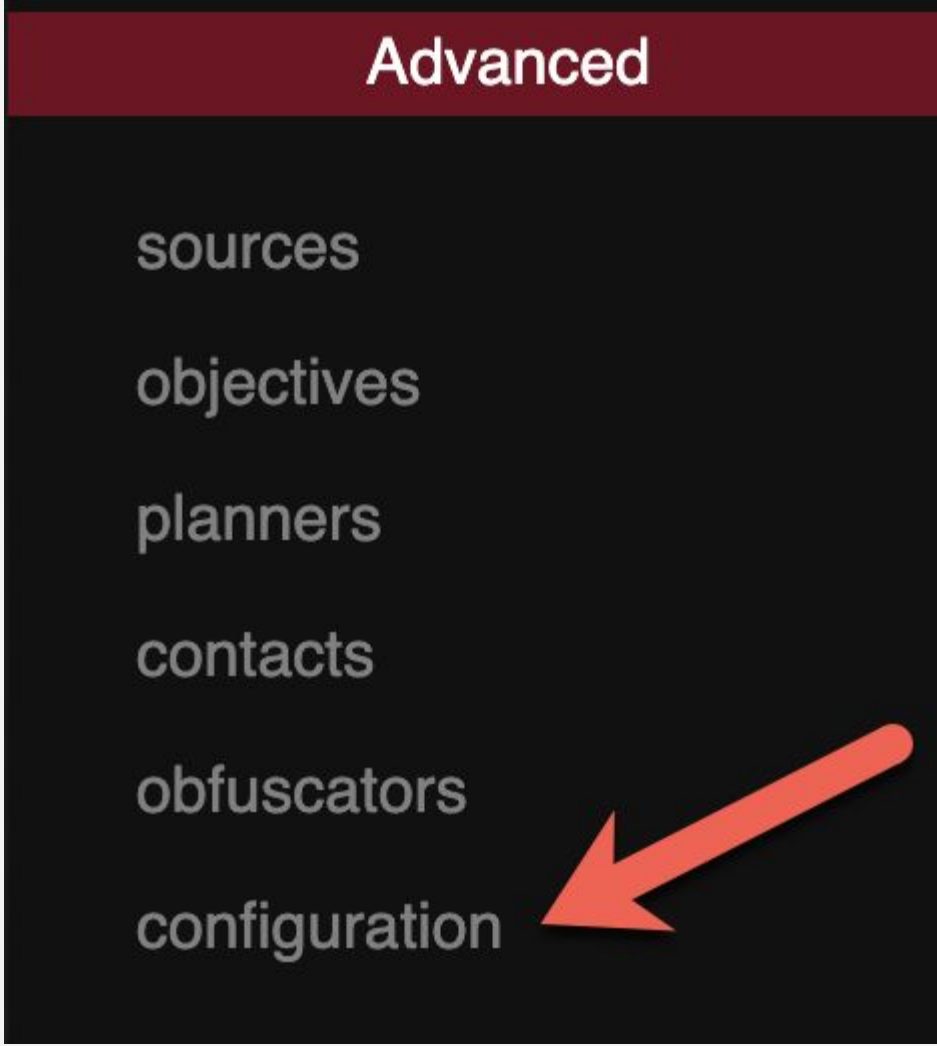


elastic

Advanced

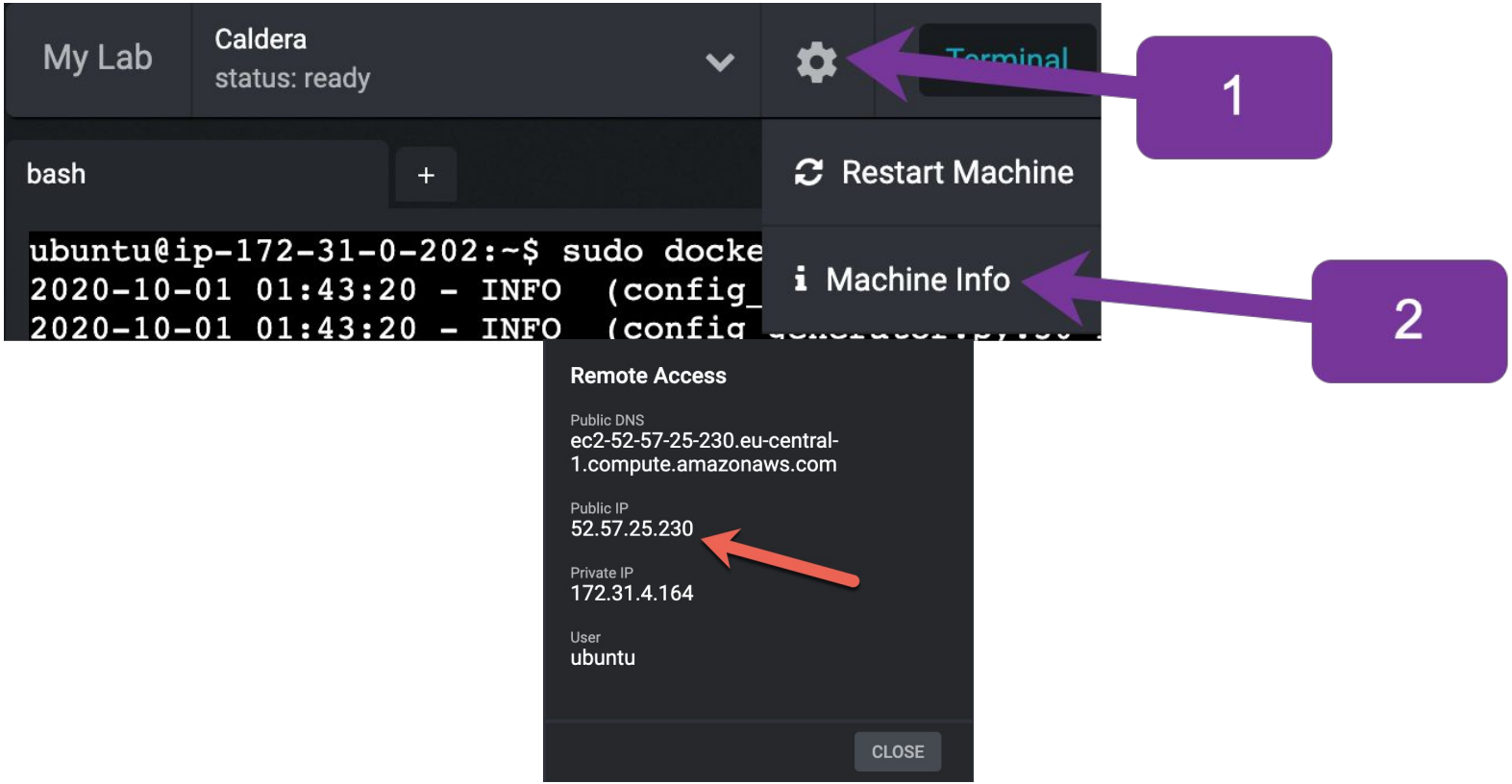
Here be dragons!

Click configuration



Retrieve IP For Caldera Server

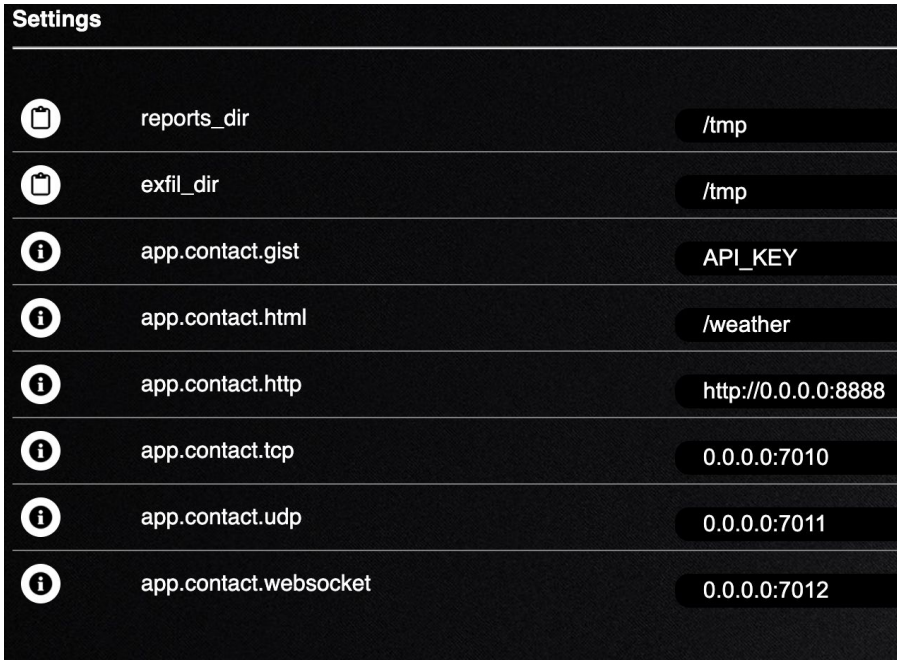
Copy the Public IP address, this will be used for the red agent

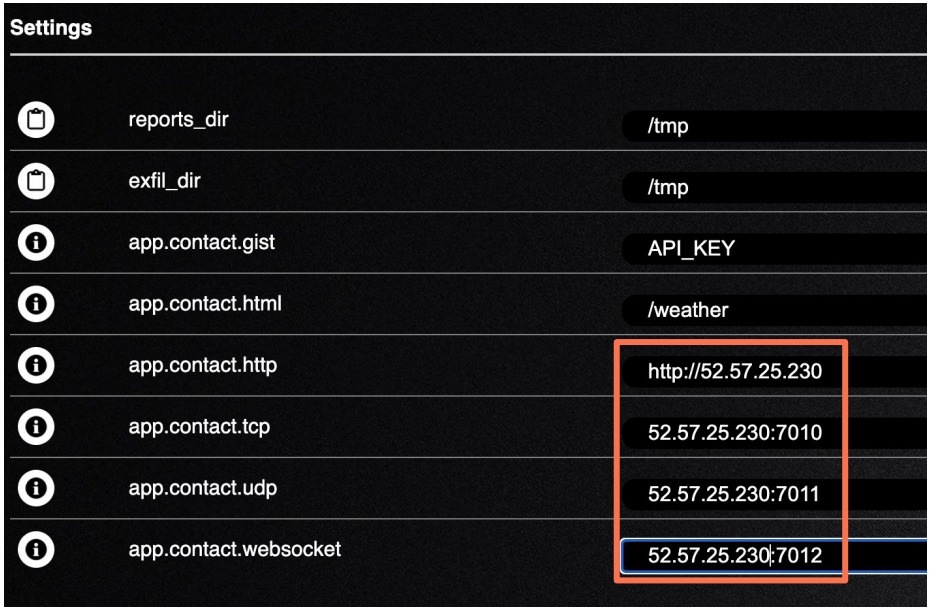


elastic

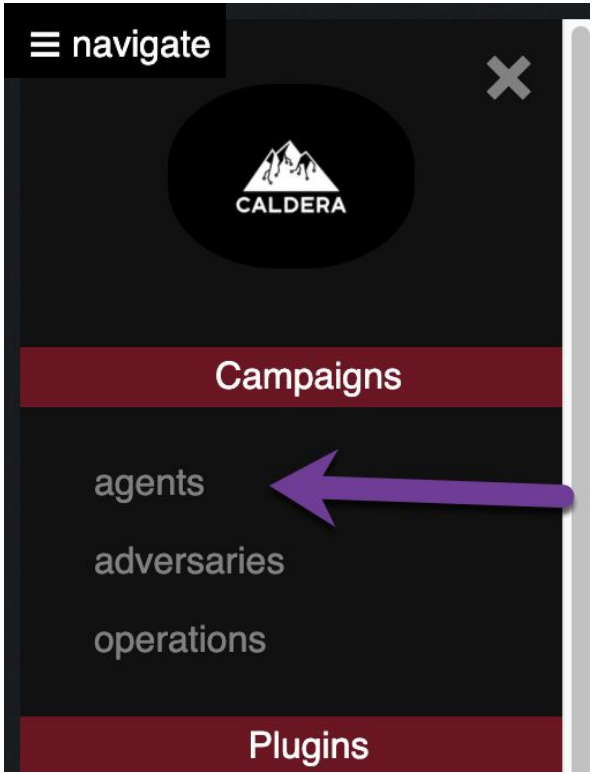
Helping our agents communicate back

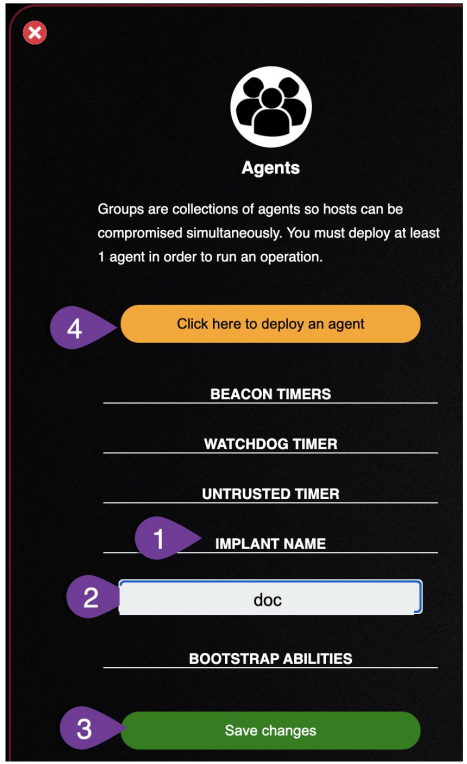
Update the **app.contact** points with the Public IP of your caldera server
Make sure to click UPDATE on the right :)





Deploying the first implant



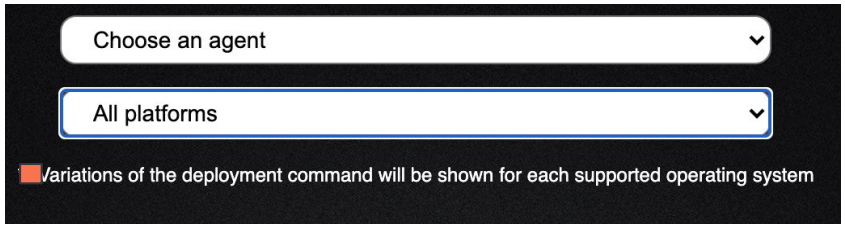


1. Click Implant Name
2. Change implant name to **doc** no extension required
3. Click to receive command you will enter on the victim machine

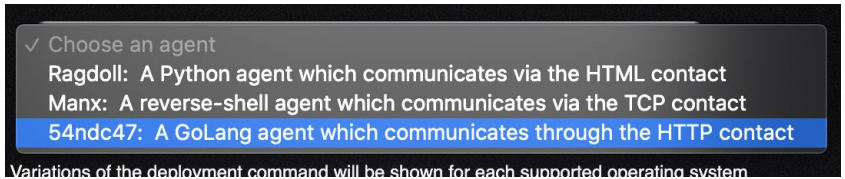
elastic

Helping your implant connect back to valid infrastructure

1



2



3

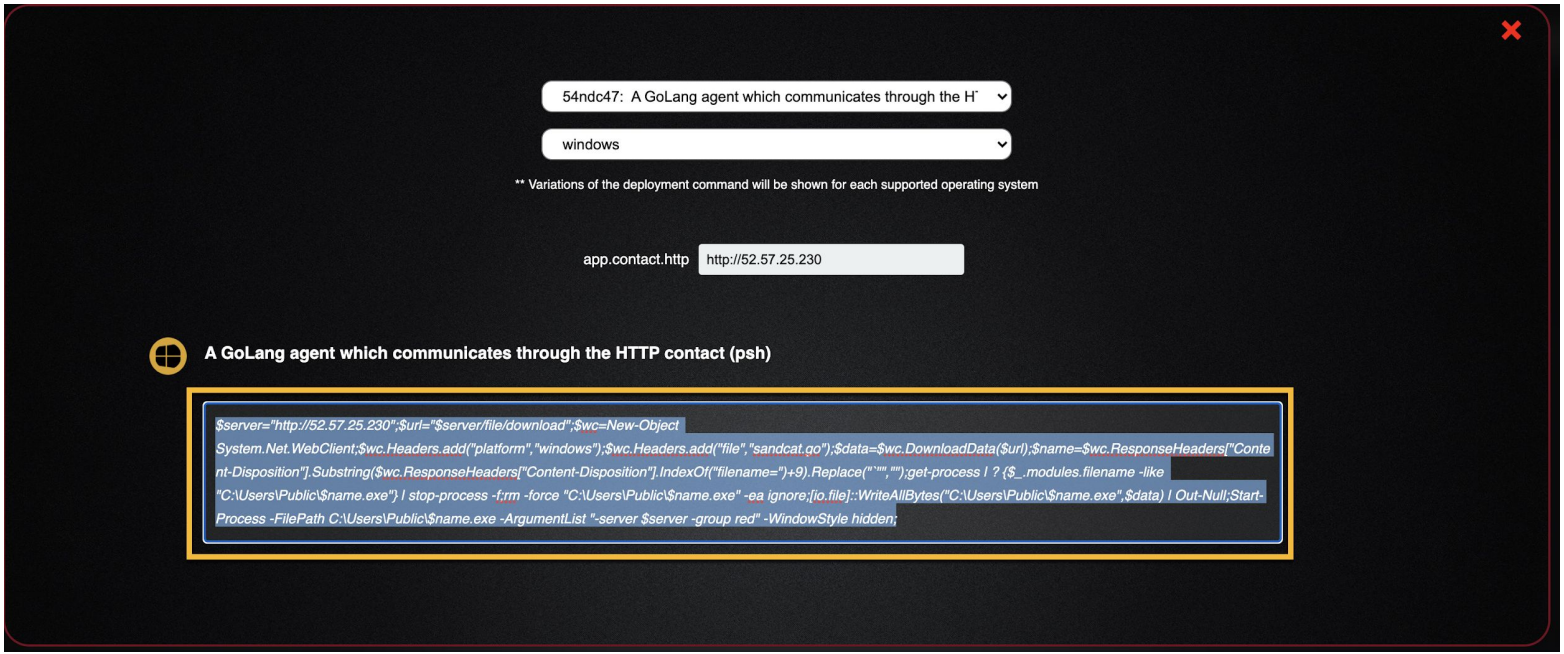
4

http://<your_public_ip>
Without the 8888 suffix

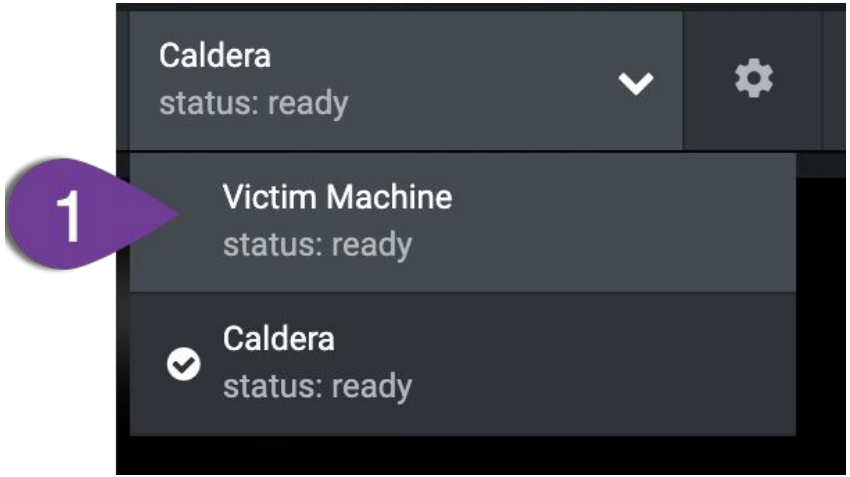
elastic

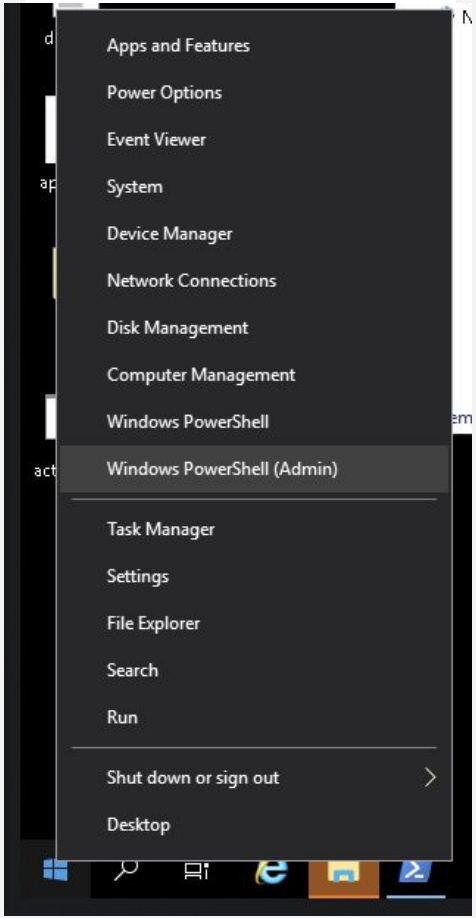
Helping your implant connect back to valid infrastructure

Implant Code Creation - **Copy the command**



Switch back to the victim machine

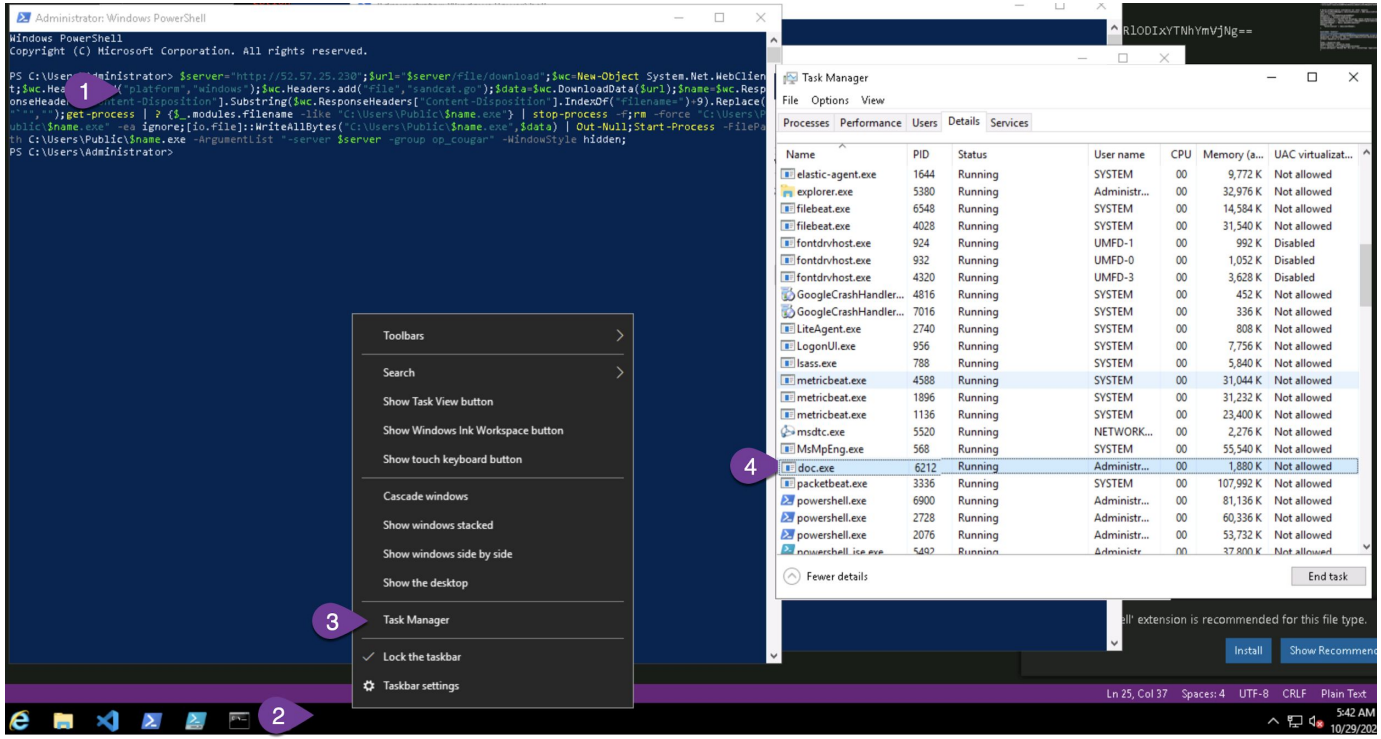




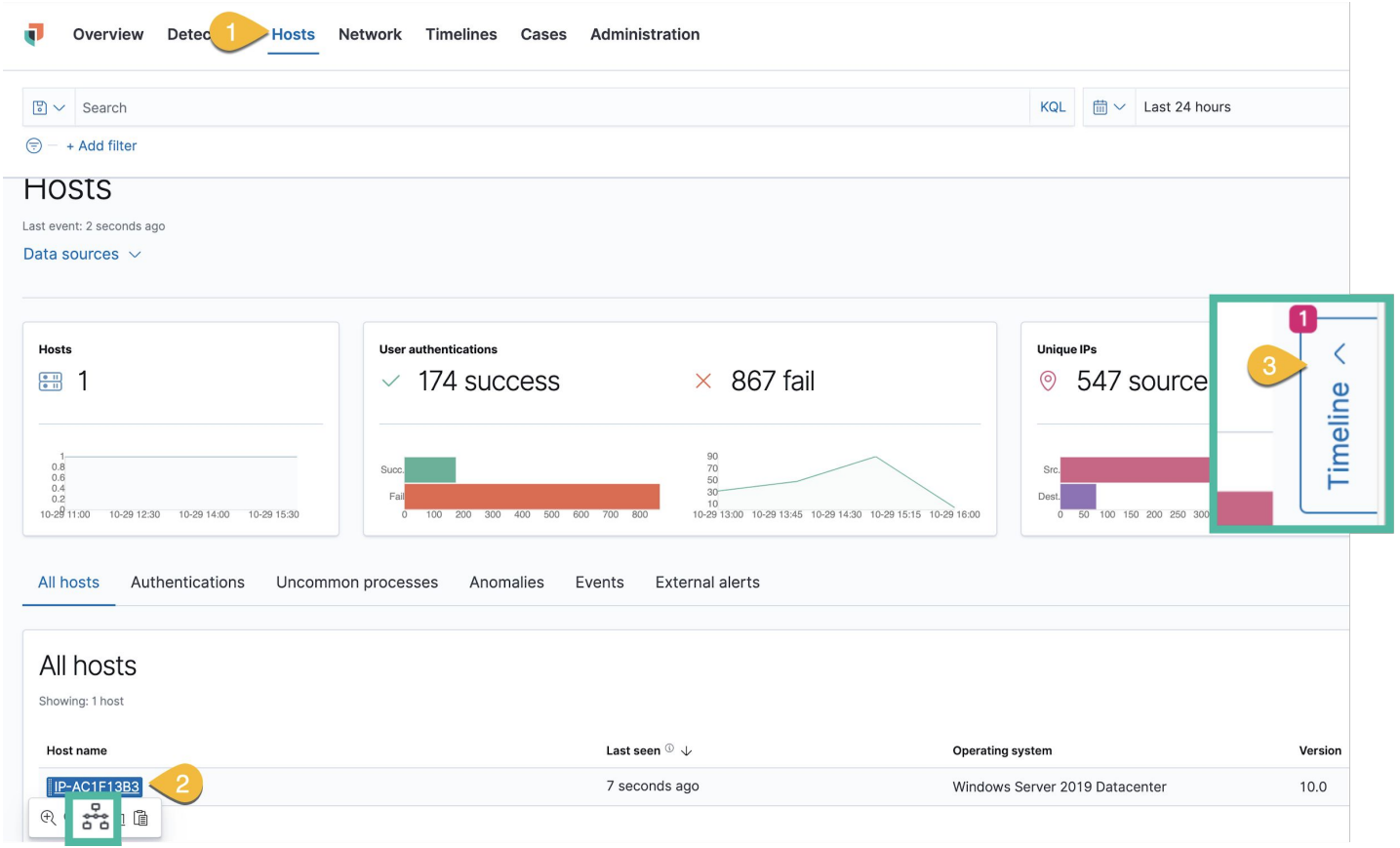
Run powershell in **admin mode** by **right clicking** the windows flag icon

Execute Powershell Script

Paste the script, execute, Open Task Manager and validate process is running



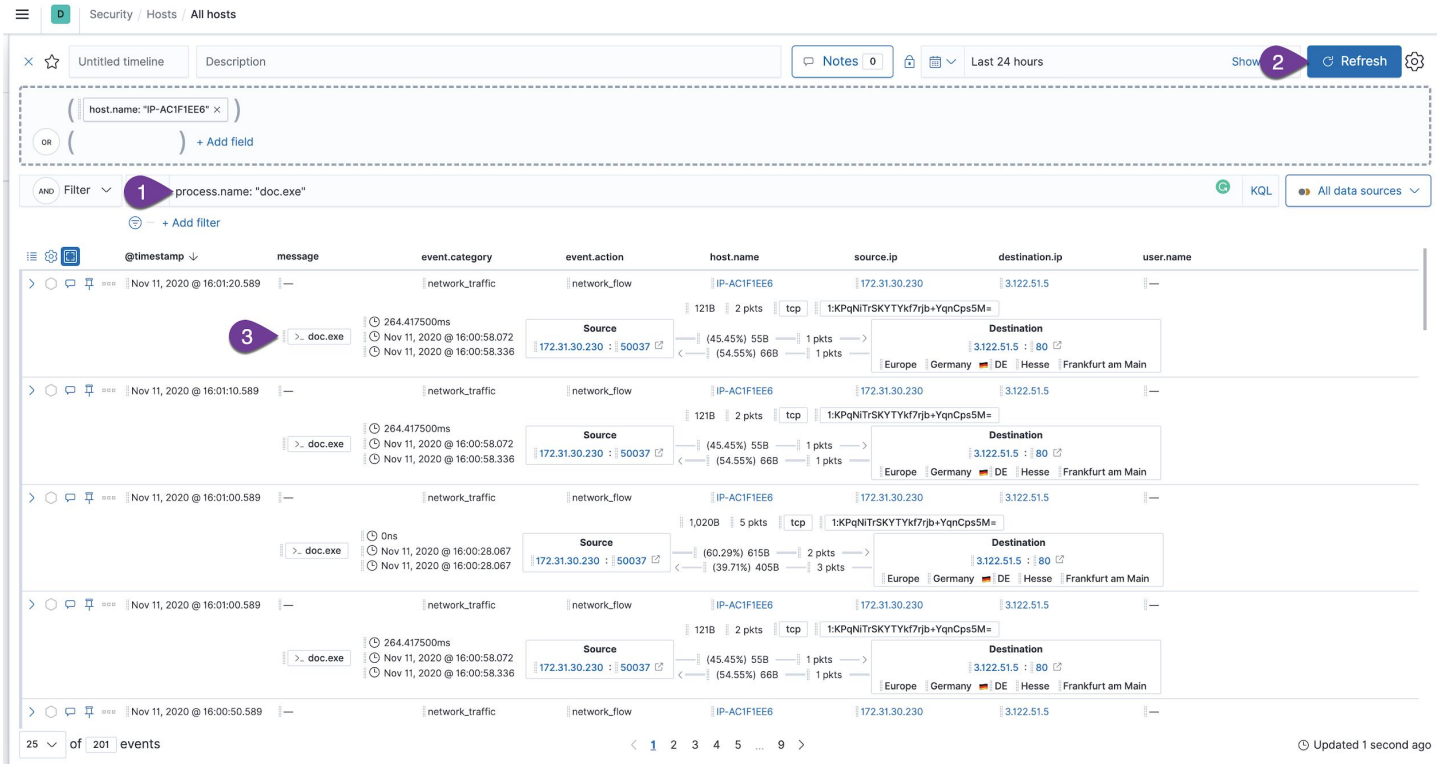
Confirm Eyes On in Elastic



elastic

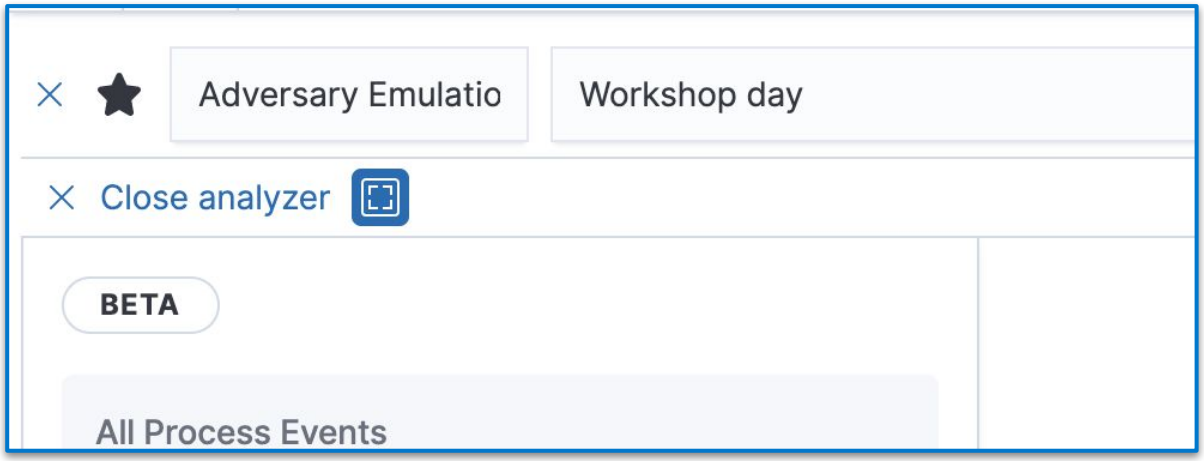
Search for the process

process.name: **doc.exe**



Save timeline

Enter a title and description. No save button required :)

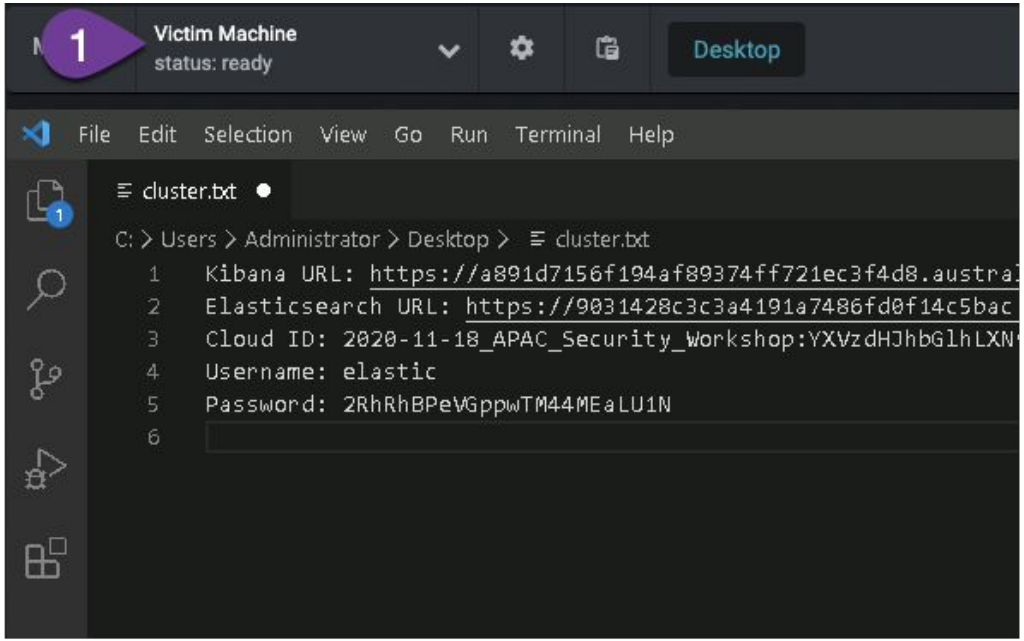


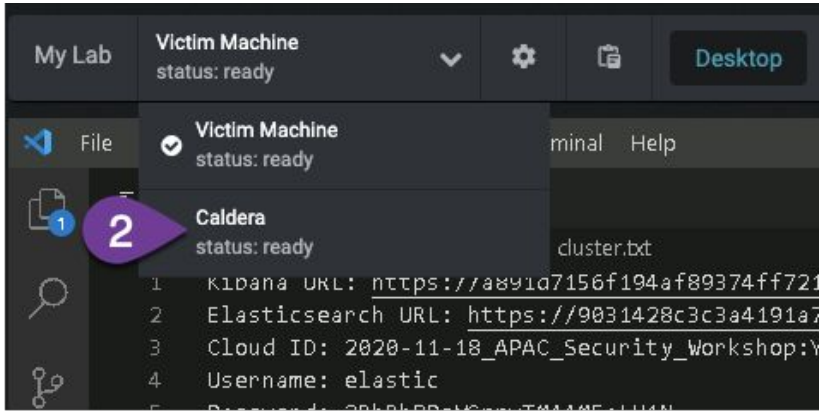
End Lab 2

Lab 3 - Adversary Execution

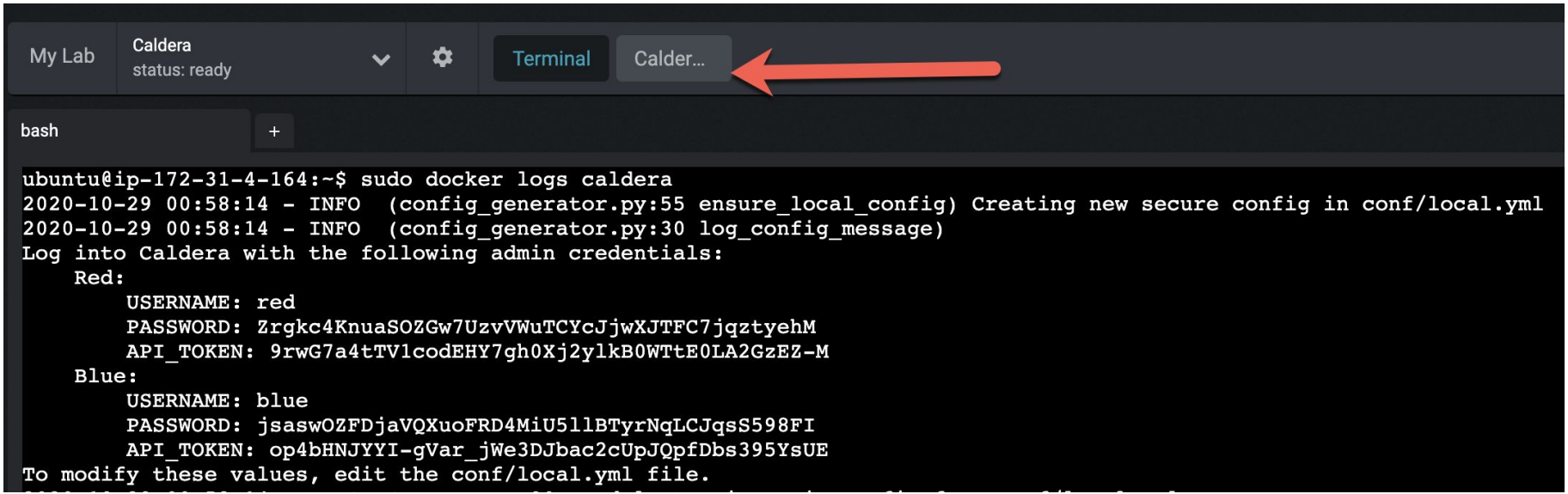
10 - 15 Minutes
Adversary sequencing and execution

Switch to Caldera



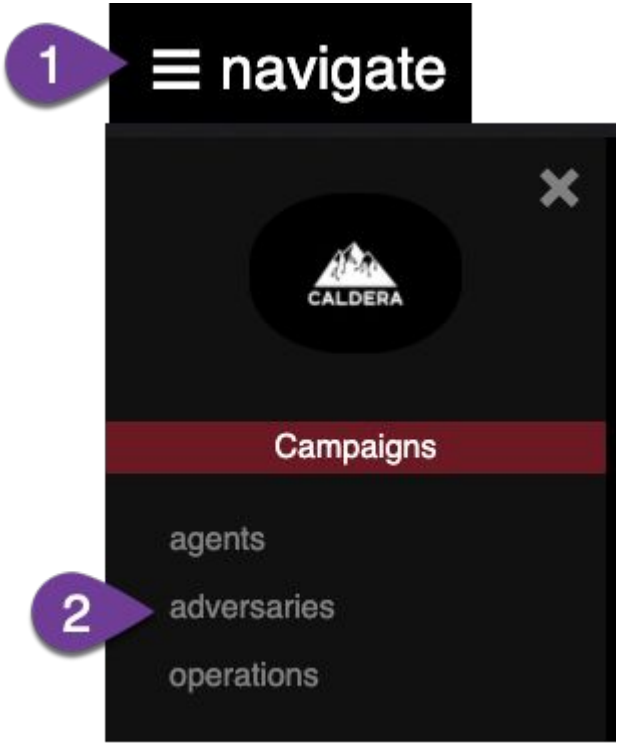


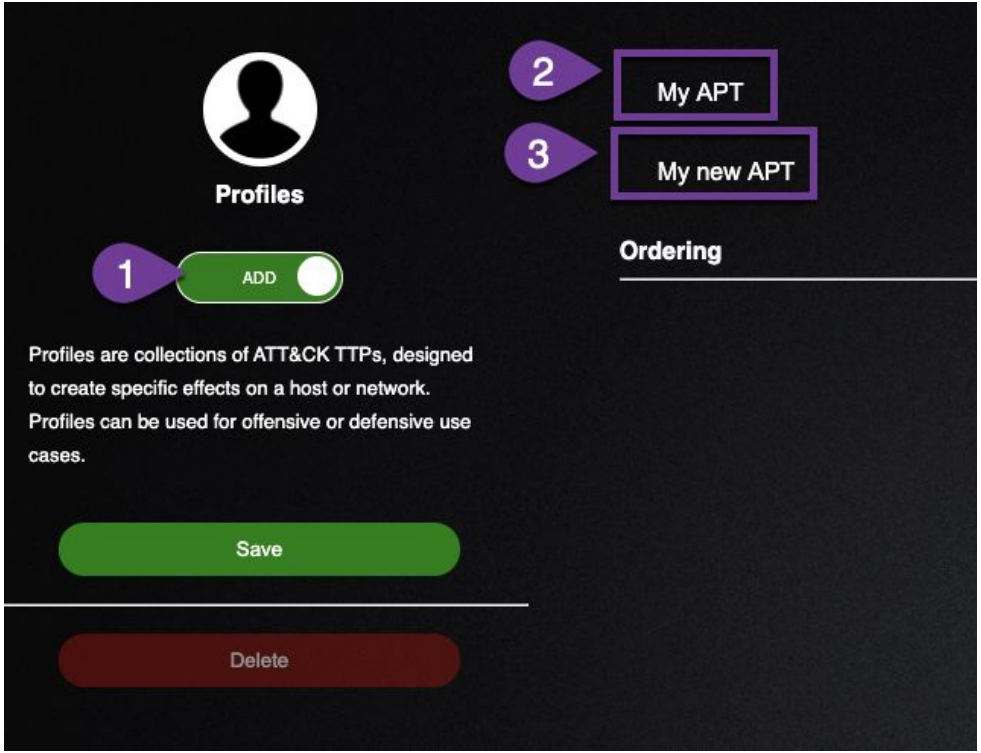
Open Caldera Web Console



Adversary Creation

Open Adversaries and create a new Adversary

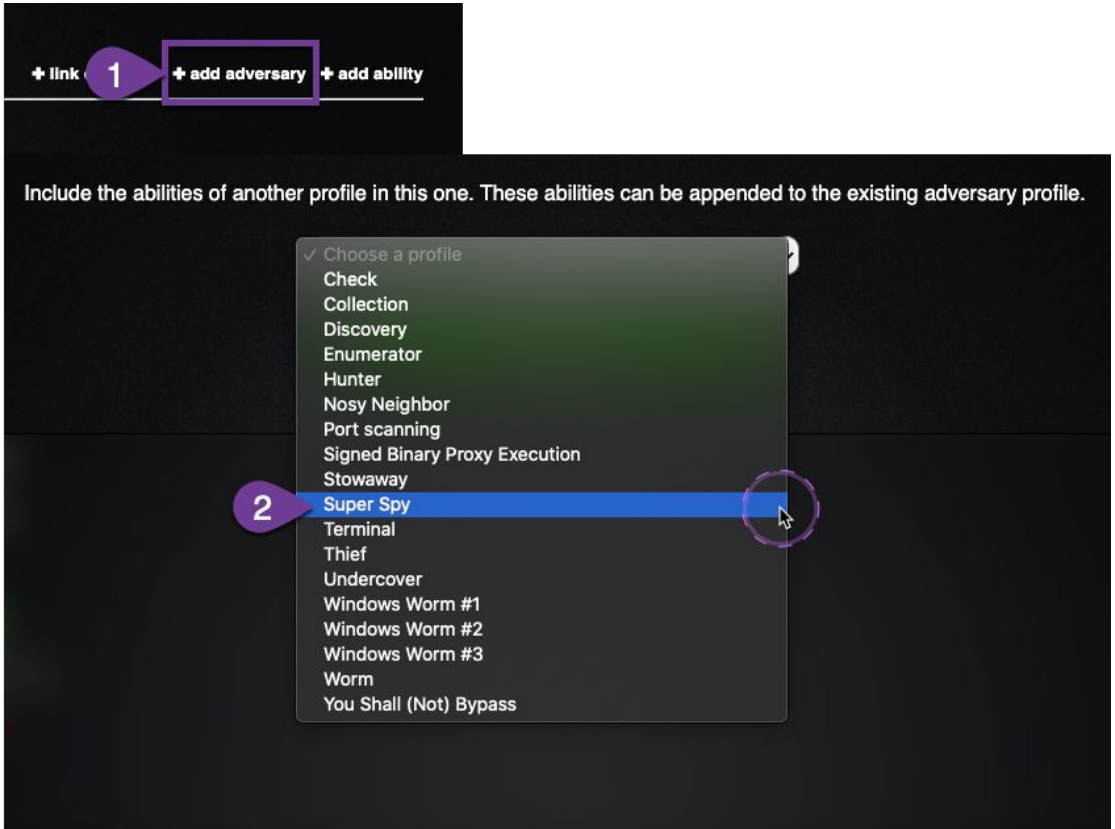




elastic

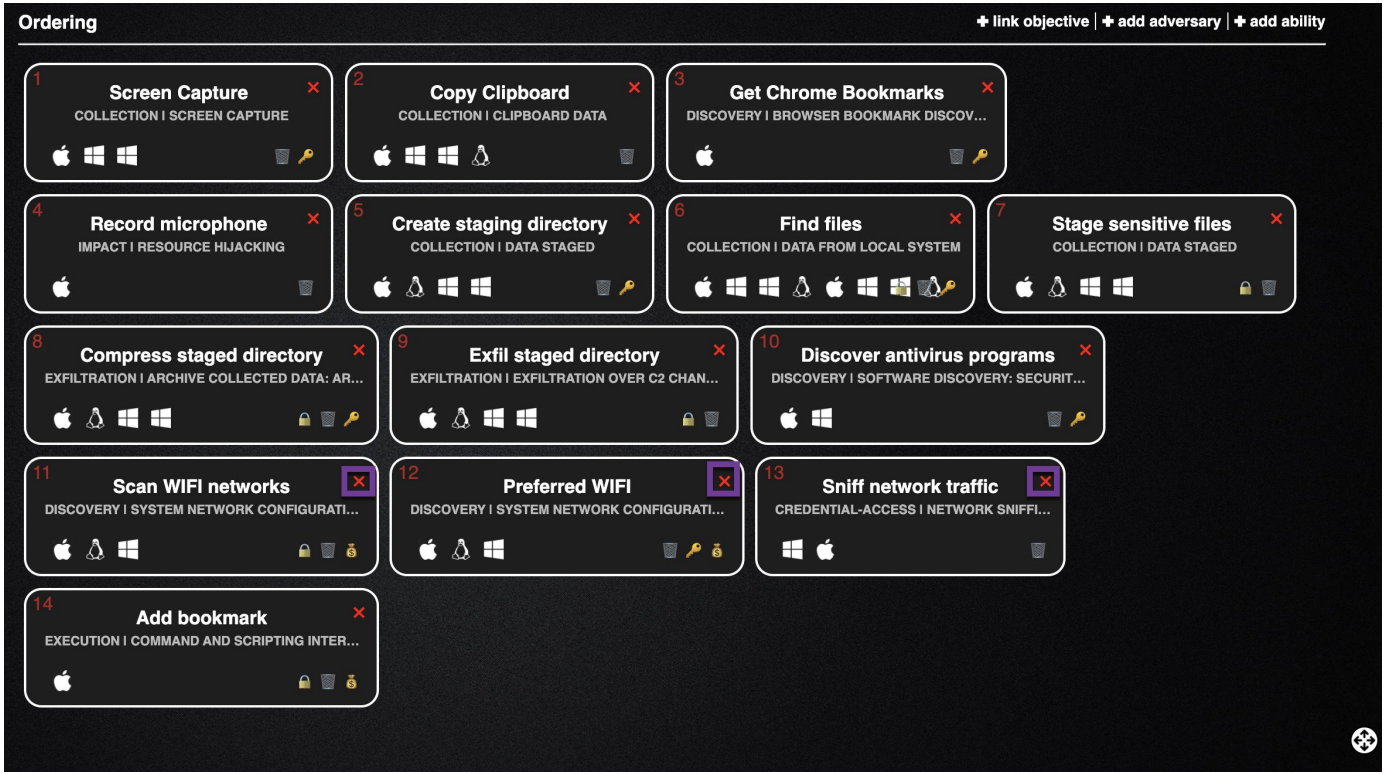
Default Adversary

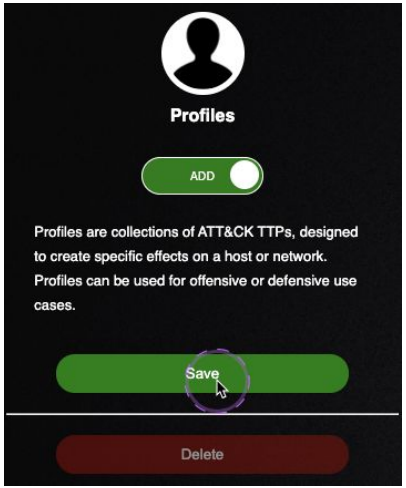
Select Add Adversary | Super Spy



Remove Irrelevant Abilities

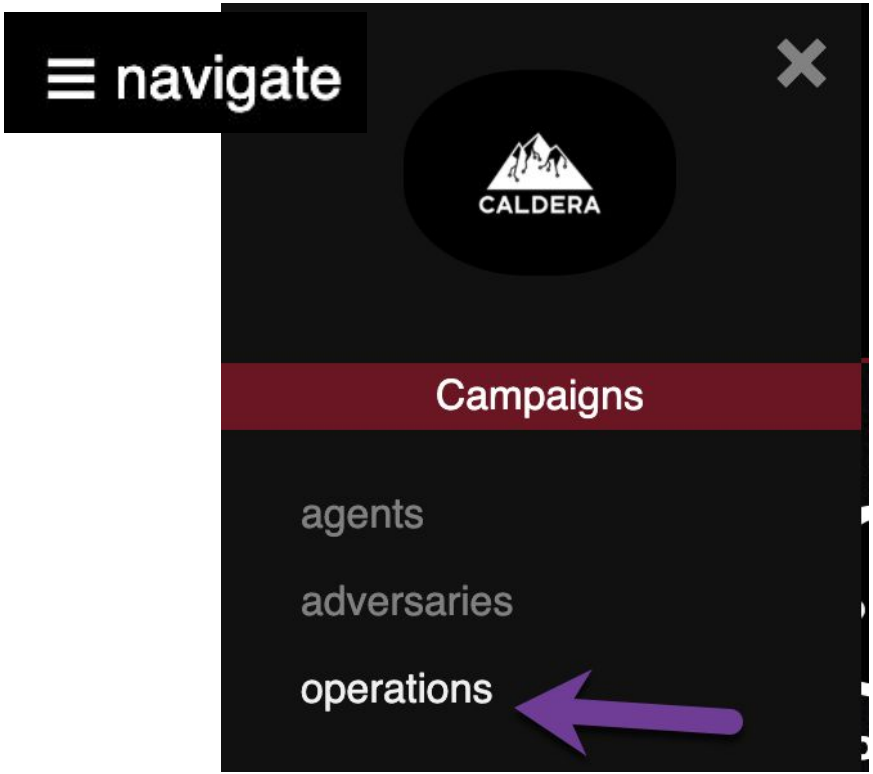
Remove **Wifi Networks** and **Sniff network traffic then save**

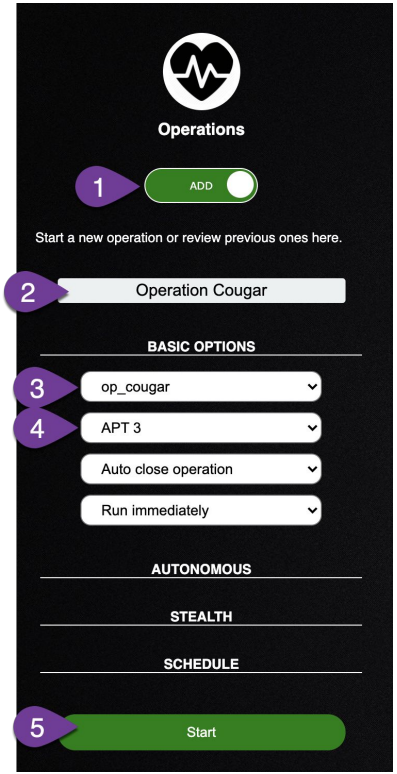




Operation Cougar Initiate

Select **Super Spy** or **APT3** as your group

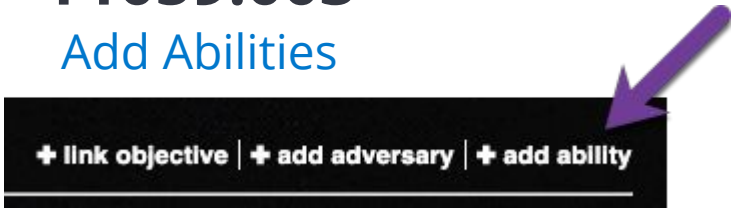


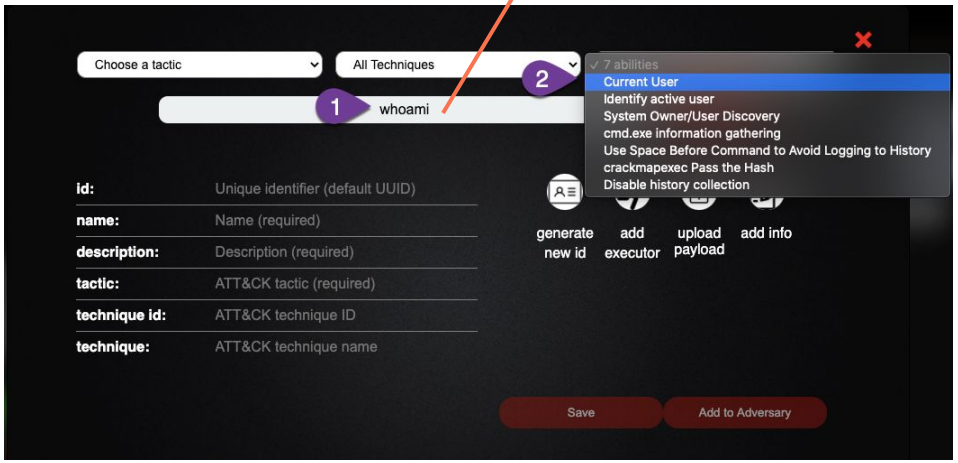


T1059.003

Add Abilities

OPTIONAL



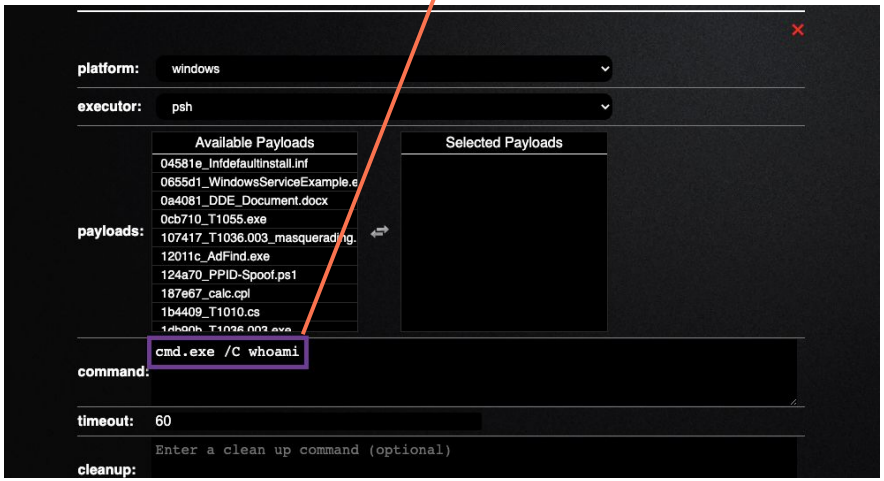


Enterprise	T1059	.003	Command and Scripting Interpreter: Windows Command Shell	An APT3 downloader uses the Windows command `"cmd.exe" /C whoami`. The group also uses a tool to execute commands on remote computers.[3][4]
		.001	Command and Scripting Interpreter: PowerShell	APT3 has used PowerShell on victim systems to download and run payloads after exploitation.[3]

T1059.003 Continued

OPTIONAL

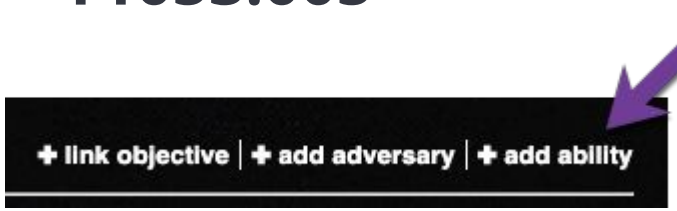
Update command to
cmd.exe /C whoami



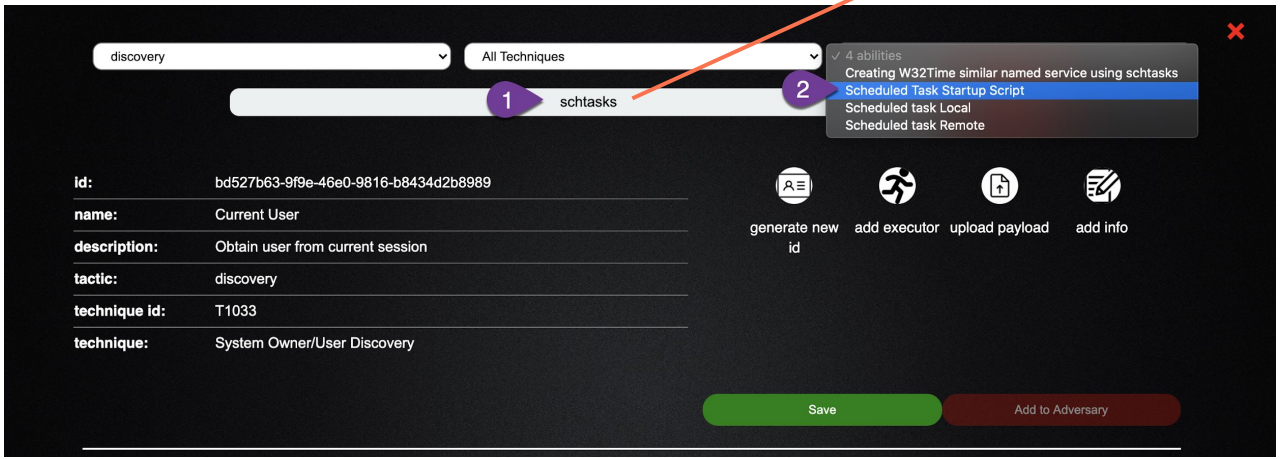
Enterprise	T1059	.003	Command and Scripting Interpreter: Windows Command Shell	An APT3 downloader uses the Windows command `"cmd.exe" /C whoami`. The group also uses a tool to execute commands on remote computers.[3][4]
		.001	Command and Scripting Interpreter: PowerShell	APT3 has used PowerShell on victim systems to download and run payloads after exploitation.[3]

T1053.005

OPTIONAL



Type
schtasks



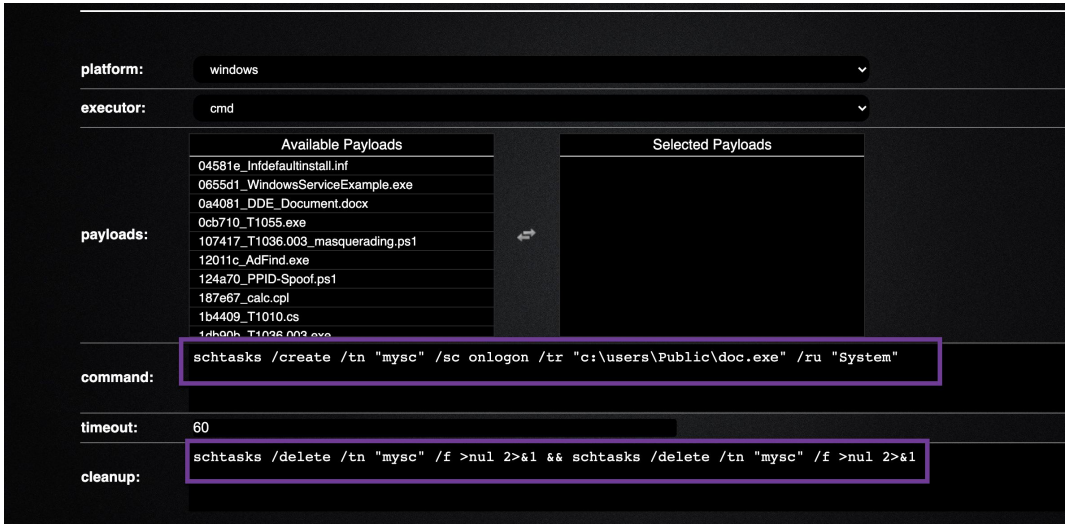
Enterprise	T1053	.005	Scheduled Task/Job: Scheduled Task	An APT3 downloader creates persistence by creating the following scheduled task: `schtasks /create /tn "mysc" /tr C:\Users\Public\test.exe /sc ONLOGON /ru "System"`

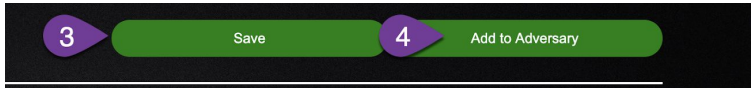
T1053.005 Continued

OPTIONAL

Command: schtasks /create /tn "mysc" /sc onlogon /tr "cmd.exe /c c:\users\public\doc.exe" && schtasks /create /tn "T1053_005_OnStartup" /sc onstart /ru system /tr "cmd.exe /c c:\users\public\doc.exe"

Cleanup: schtasks /delete /tn "mysc" /f >nul 2>&1 && schtasks /delete /tn "mysc" /f >nul 2>&1



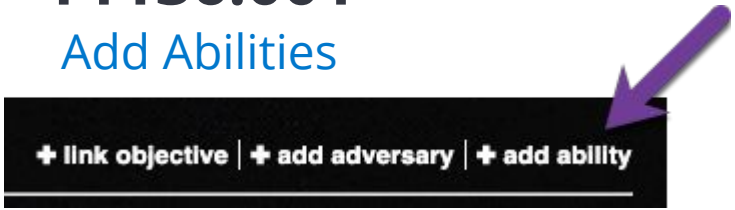


Enterprise	T1053	.005	Scheduled Task/Job: Scheduled Task	An APT3 downloader creates persistence by creating the following scheduled task: `schtasks /create /tn "mysc" /tr C:\Users\Public\test.exe /sc ONLOGON /ru "System"`.

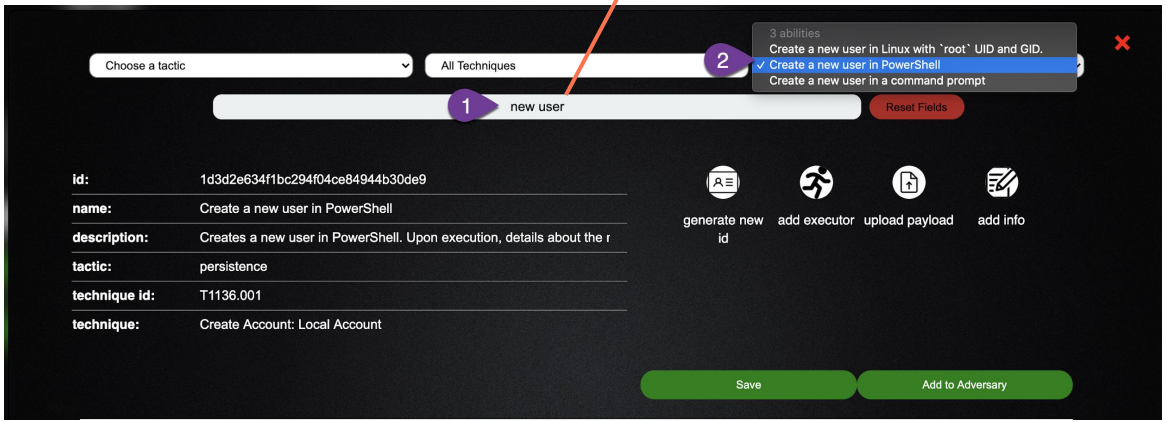
T1136.001

Add Abilities

OPTIONAL



Type
new user



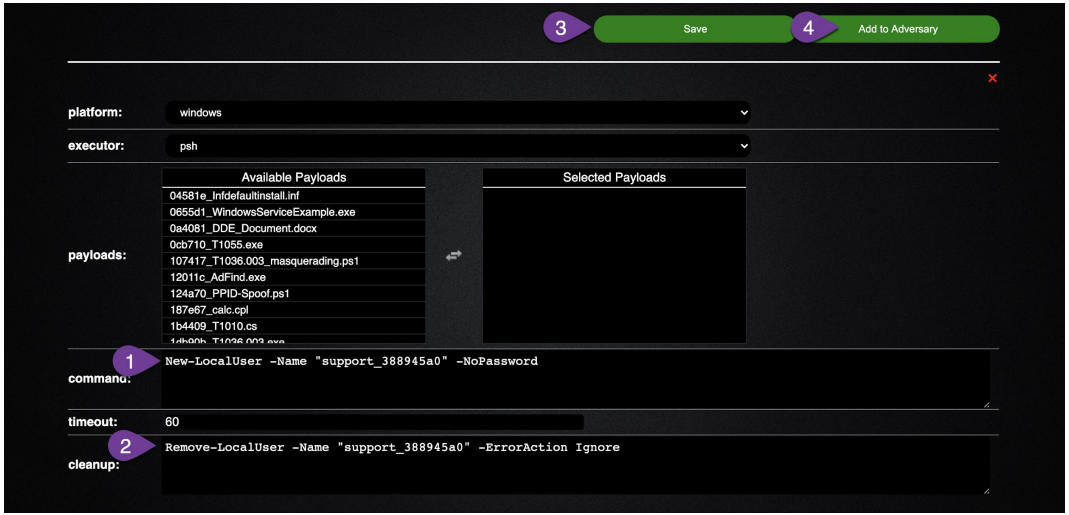
Enterprise	T1136	.001	Create Account: Local Account	APT3 has been known to create or enable accounts, such as `support_388945a0`.[7]

T1136.001 Continued

OPTIONAL

Command: **New-LocalUser -Name "support_388945a0" -NoPassword**

Cleanup: **Remove-LocalUser -Name "support_388945a0" -ErrorAction Ignore**



Enterprise	T1136	.001	Create Account: Local Account	APT3 has been known to create or enable accounts, such as `support_388945a0`. [7]

End Lab 3

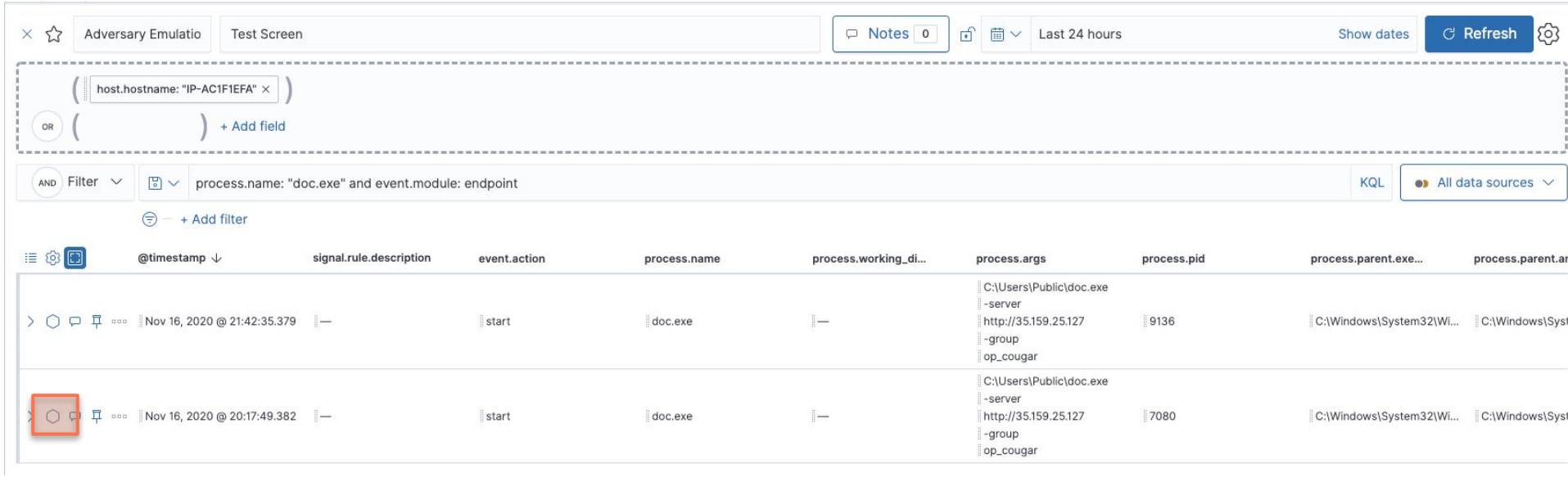
Lab 4 - Detection

Detection

Review dataset

Look at the last event by time

Filter: **process.name: “doc.exe” and event.module: “endpoint”**

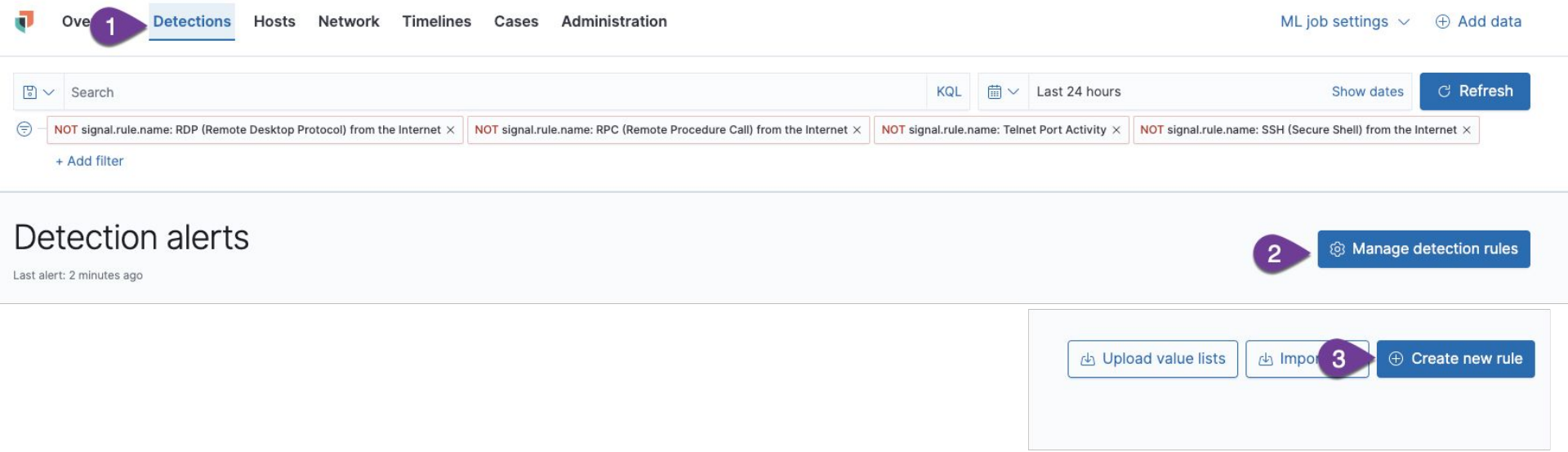


elastic

Process Tree View

Your view may be different

Create First EQL Rule



Command & Control (C2)

Looking for LOLBIN downloading an executable

Detection:

- OS signed binaries used by attackers to evade detection
  - PowerShell, VB script, cmd.exe

EQL:

```
sequence by host.name
  [process where event.type == "start"
    and process.name in ("powershell.exe", "mshta.exe", "installutil.exe",
"msxsl.exe", "rundll32.exe") ]
[file where file.extension == "exe"
[process where process.code_signature.status != "trusted" and event.type==
"start"
  [network where true]
```

elastic

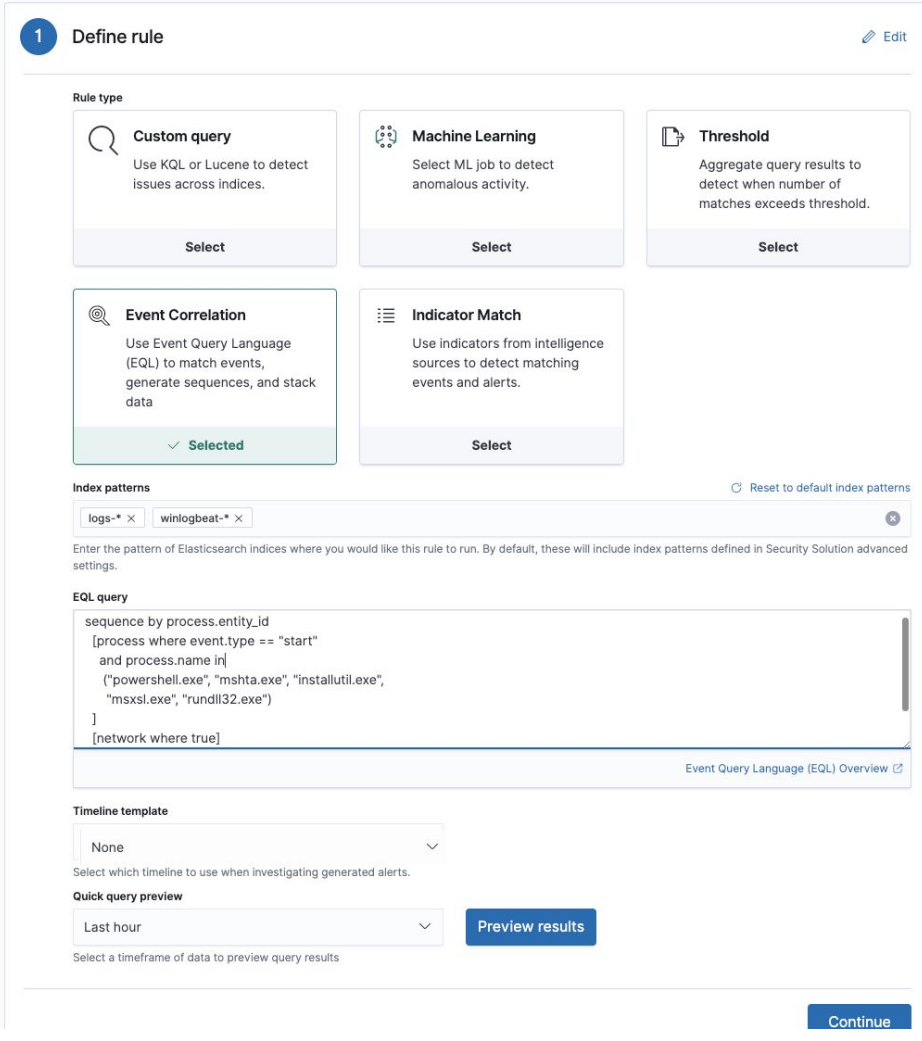
Process

Define Rule

- Select “Event Correlation”
- Remove all index patterns except “logs-*” and “winlogbeat-*”
- Copy query

```
sequence by host.name
  [process where event.type == "start"
    and process.name in
     ("powershell.exe", "mshta.exe", "installutil.exe",
      "msxsl.exe", "rundll32.exe")
  ]
[file where file.extension == "exe"]
[process where process.code_signature.status !=
"trusted" and event.type == "start"]
[network where true]
```

- Quick query preview of last 24 hours
- Press continue (don’t preview results just yet)



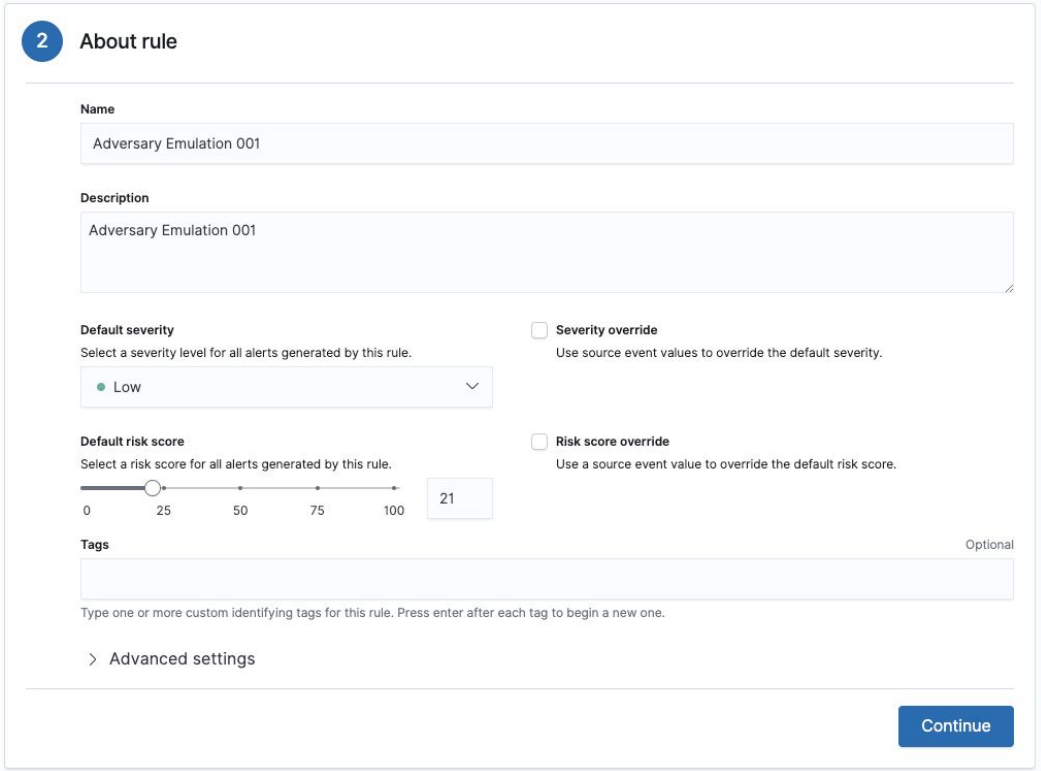
Process Continued

About Rule

Name: Adversary Emulation 001

Description: Same

Press continue



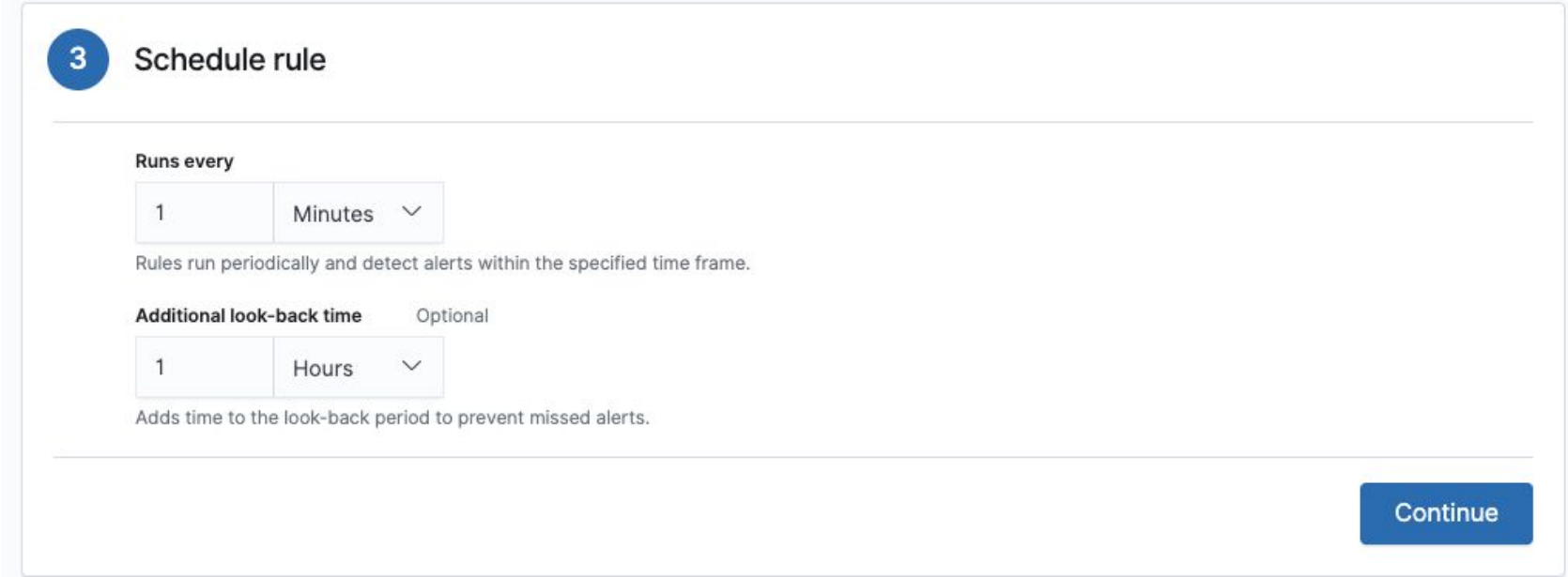
elastic

Process Continued

Schedule Rule

Runs Every: **1 Minutes**

Lookback: **1 Hours**

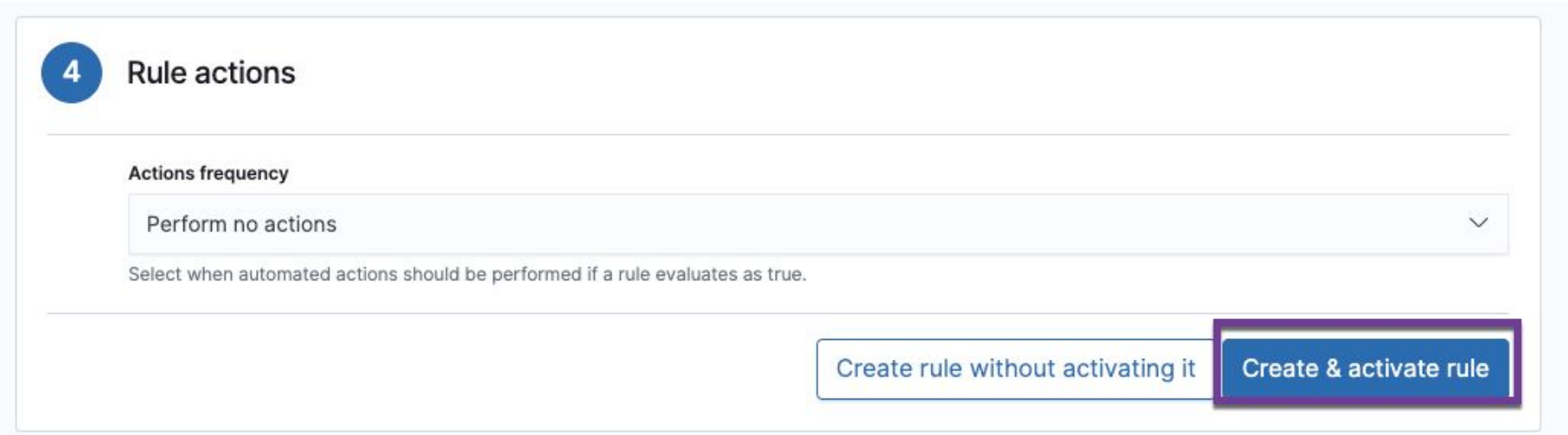


elastic

Process Continued

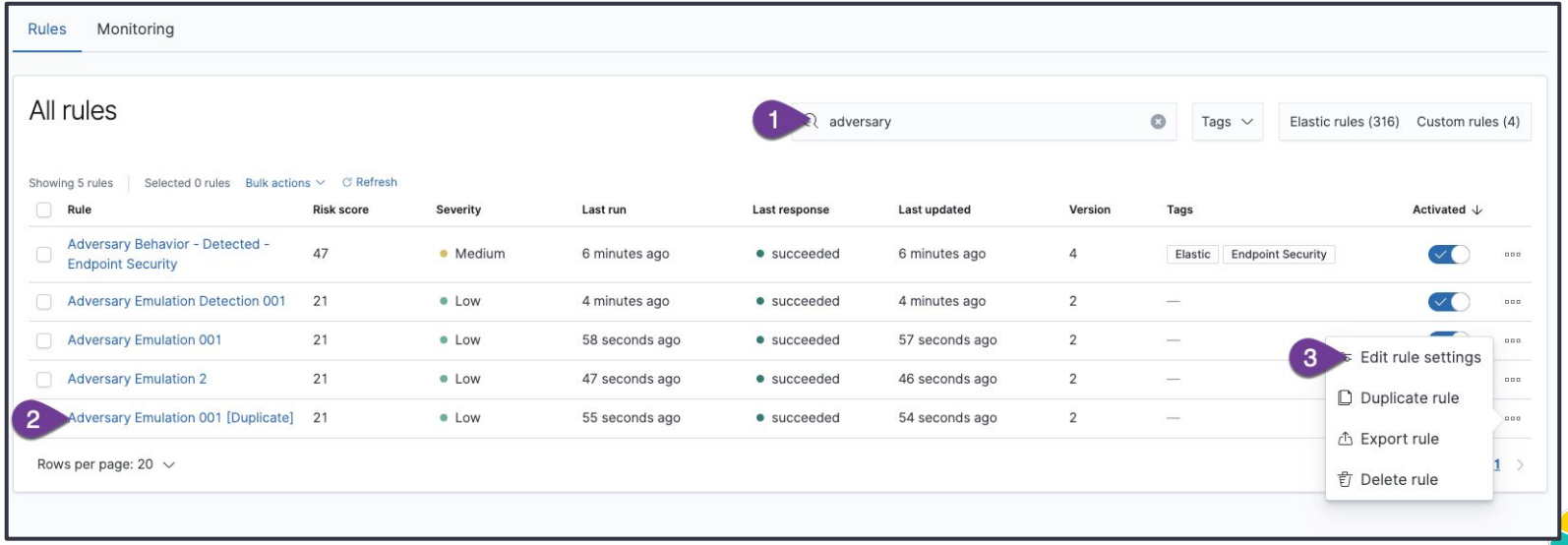
Rule Actions

Create & activate rule



Let's improve the rule a little bit

Duplicate the rule



elastic

Command & Control (C2)

Looking for LOLBIN Network Activity (TXXXX)

Addition:

- Detect where a file has been written by powershell.exe
- Until conhost.exe is run.

EQL:

```
sequence by host.name
  [process where event.type == "start"
    and process.name in ("powershell.exe", "mshta.exe",
"installutil.exe",  "msxsl.exe", "rundll32.exe") ]
[file where file.extension == "exe"
[process where process.code_signature.status !=  "trusted" and
event.type== "start"
  [network where true]
  until
[ process where process.name == "conhost.exe"]
```

elastic

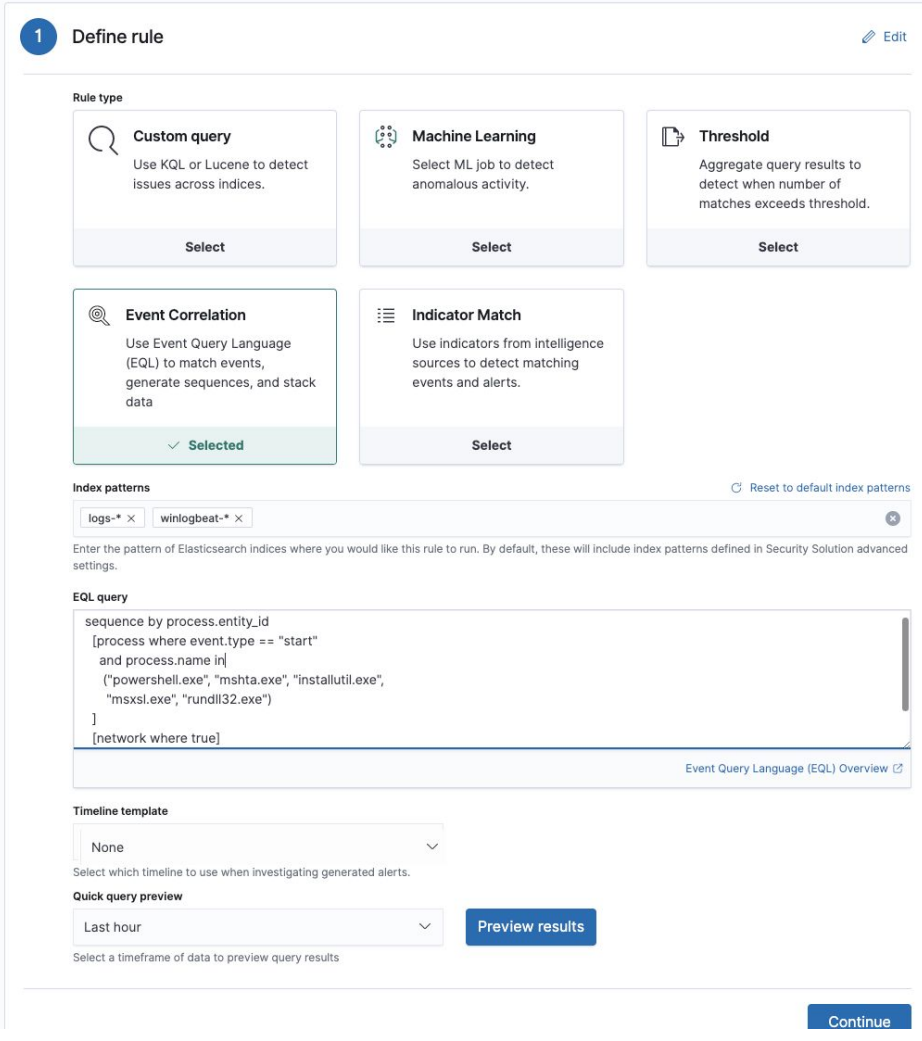
Process

Define Rule

- Select “Event Correlation”
- Remove all index patterns except “logs-*” and “winlogbeat-*”
- Copy query

```
sequence by host.name
  [process where event.type == "start"
    and process.name in ("powershell.exe", "mshta.exe",
"installutil.exe",  "msxsl.exe", "rundll32.exe") ]
[file where file.extension == "exe"
[process where process.code_signature.status !=
"trusted" and event.type== "start"
  [network where true]
  until
[ process where process.name == "conhost.exe"]
```

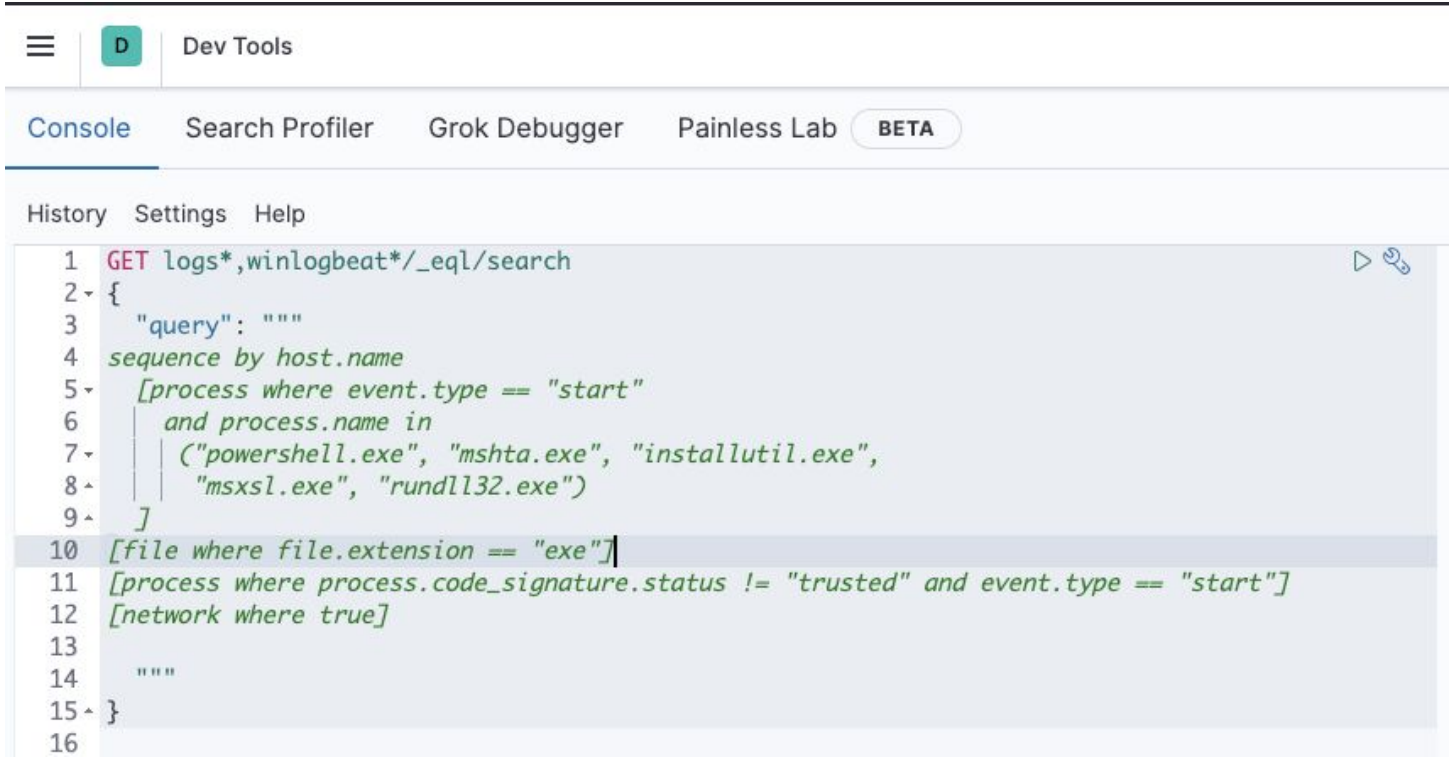
- Quick query preview of last 24 hours
- Press continue (don’t preview results just yet)



Test your detection in dev tools

GET logs*,winlogbeat*/_eql/search

OPTIONAL



End Lab 4