# Advanced Emulation Lab

# Table of contents



# List of Figures



# List of Tables

# 1. Lab Objective

- Emulate APT29 phishing (T1566.001)
- Deliver payloads and achieve persistence.

# 2. Tools

- Caldera (orchestration / emulation)
- Metasploit (payloads / post-exploitation)
- PyPhisher (phishing site / credential harvesting)

# 3. Lab Setup

- ***Kali : 192.168.1.48*** -Attacker (caldera,metasploit,pyphsisher)
- ***Windows 10 : 192.168.1.46 -*** Victim

# 4. Methodology

***Step 1:*** Create spear-phish using PyPhisher.

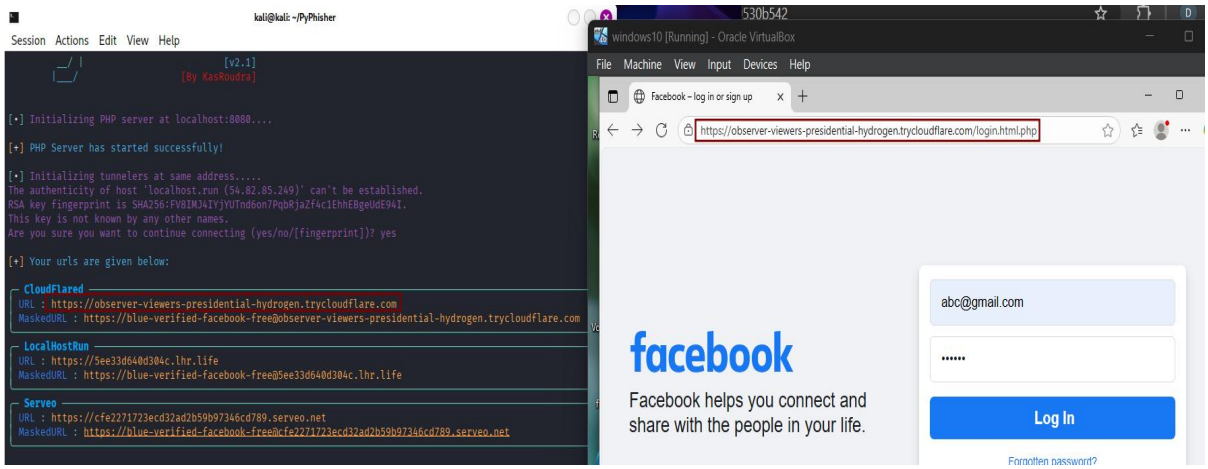



*Figure 3.1 Shows phishing link being opened by victim*

***Step 2:*** Host credential capture page and track clicks.

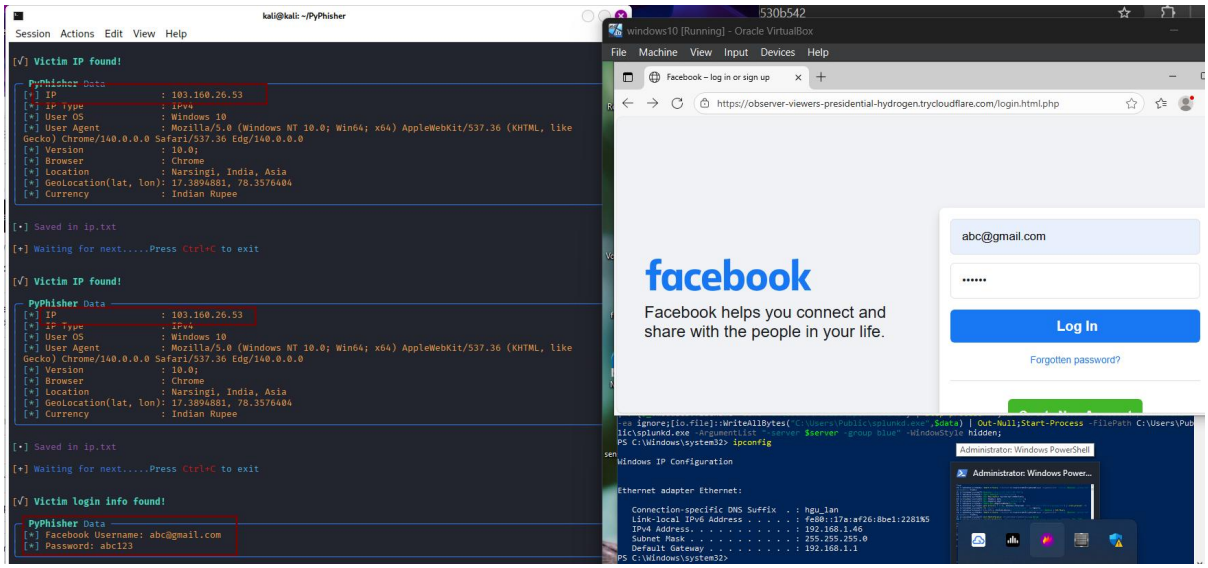



*Figure 3.2 Shows victim details being captured*

***Step 3:*** Deliver Metasploit payloads to compromised host using msfvenom Meterpreter session is successfully created

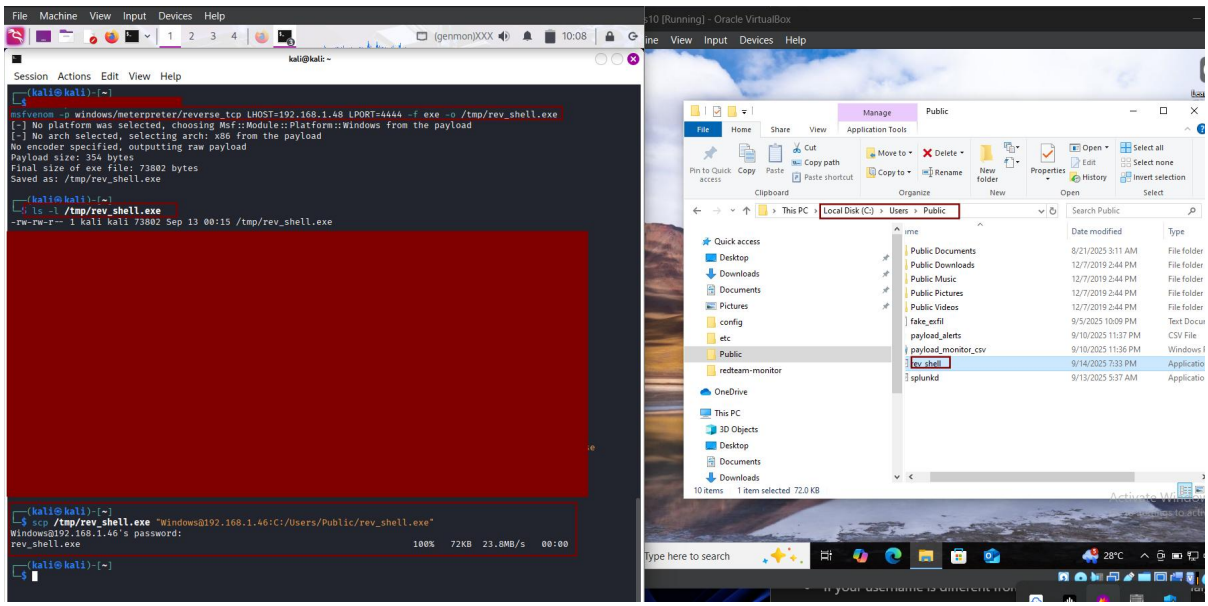



*Figure 3.3 Shows payload being sent to windows machine*

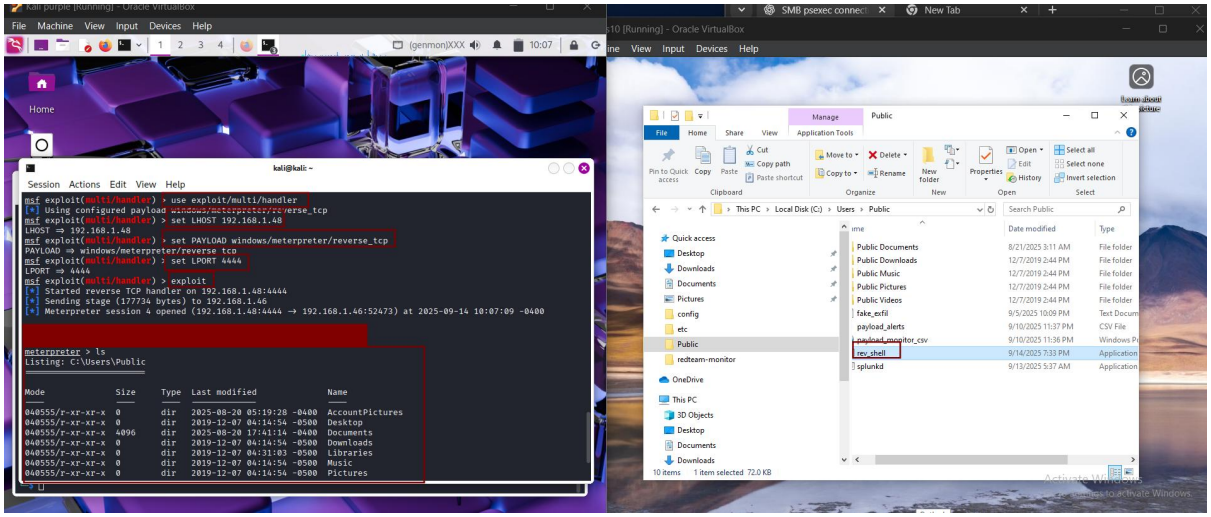



*Figure 3.4 Shows meterpreter session beig opened in kali*

***Step 4:*** Use Caldera for Adversary Emulation and login with red caldera with windows 10 using sand-cat agent

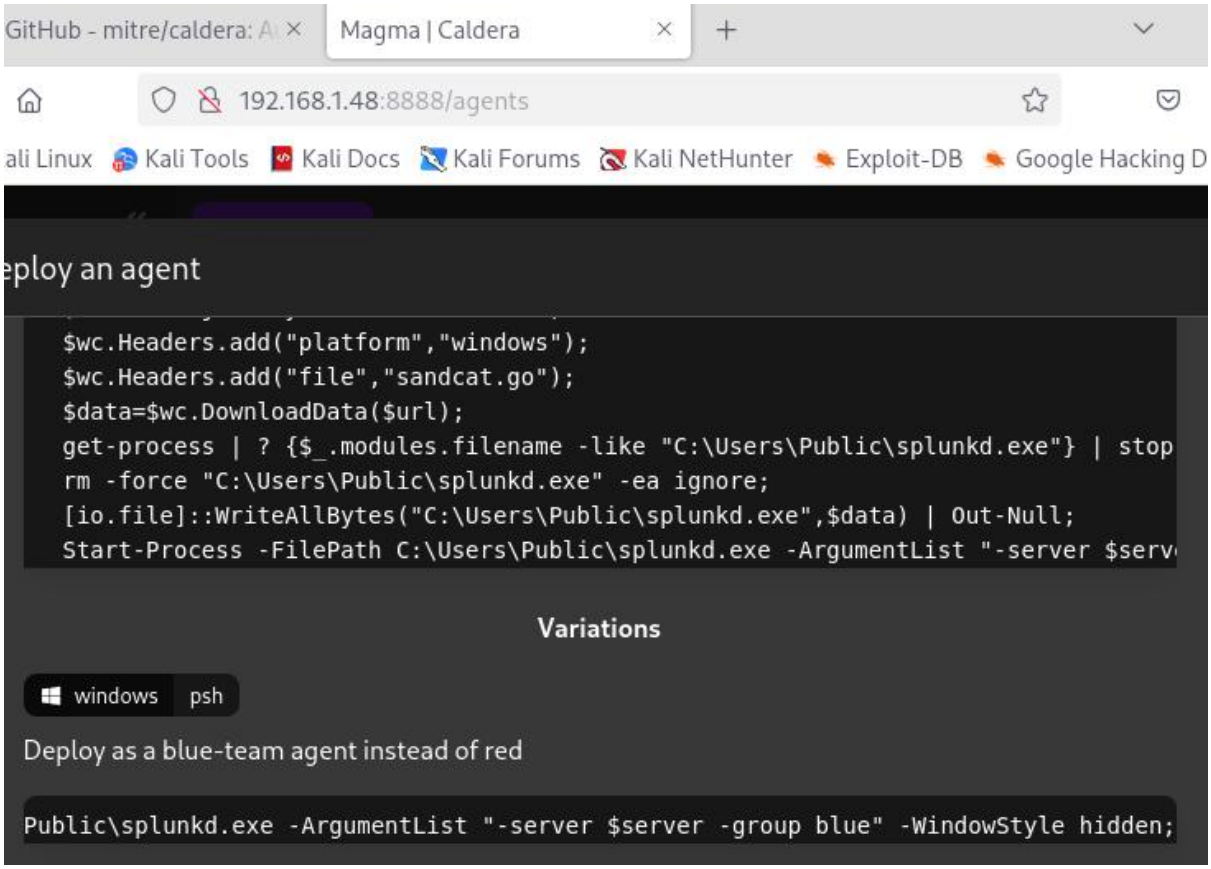



*Figure 3.5 Shows agent deployment payload*

***Step 5:*** Copy the code for red team agent as paste on windows 10

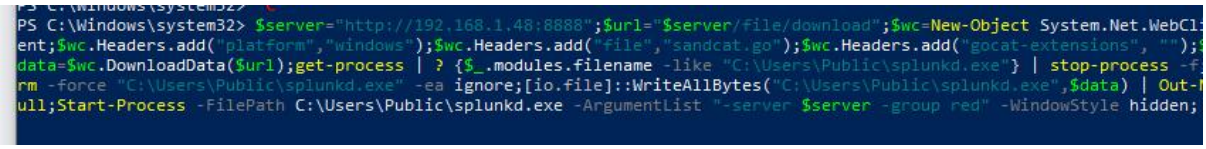



*Figure 3.6 Shows payload being pasted on victim machine*

***Step 6:*** We see the agent successfully present in our red caldera

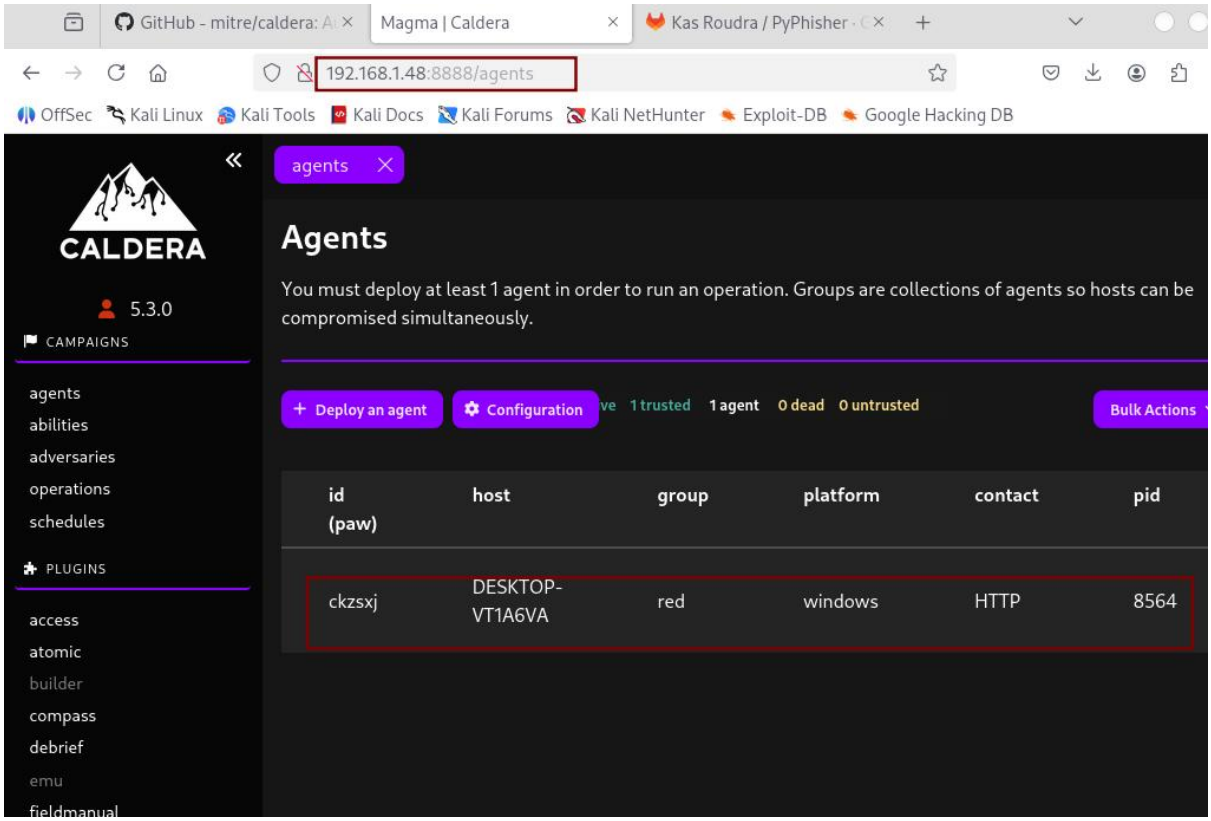



*Figure 3.7 Shows agent being successfully deployed on caldera*

***Step 7:*** Once we have our agent ,lets start our emulation ,

We are making use of below abilities

- Download Macro-Enabled Phishing Attachment
- Create a Process using WMI Query and an Encoded Command
- Winlogon HKLM Shell Key Persistence – PowerShell
- Identify local users
- Zip a Folder with PowerShell for Staging in Temp
- Exfiltrating Hex-Encoded Data Chunks over HTTP

***Step 8:*** Start making necessary changes to the ability: ***Download Macro-Enabled Phishing Attachment*** and save it.

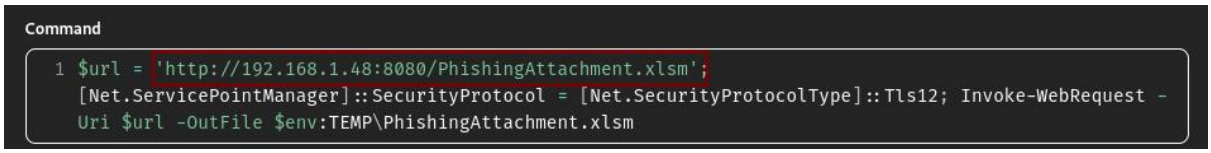



*Figure 3.8 Shows making changes to macro phishing attachment*

***Step 9:*** For ability ***Zip a Folder with PowerShell for Staging in Temp*** make the following changes and save it

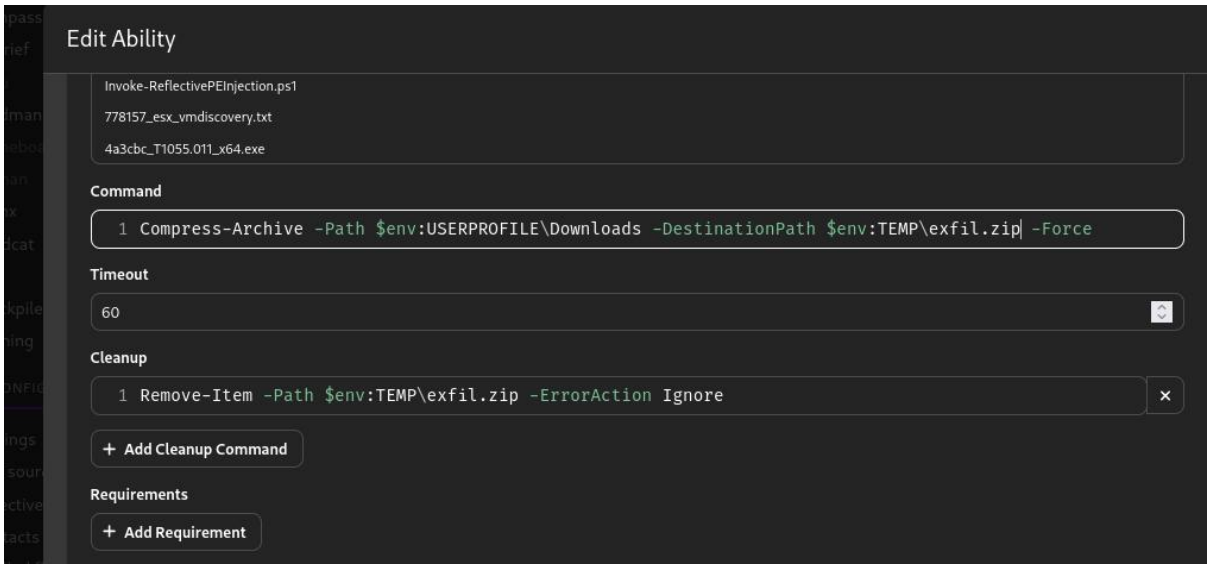



*Figure 3.9 Shows making changes to zip a folder ability*

***Step 10:*** Exfiltrating Hex-Encoded Data Chunks over HTTP

Since this ability is not present we create a new ability and make the following changes

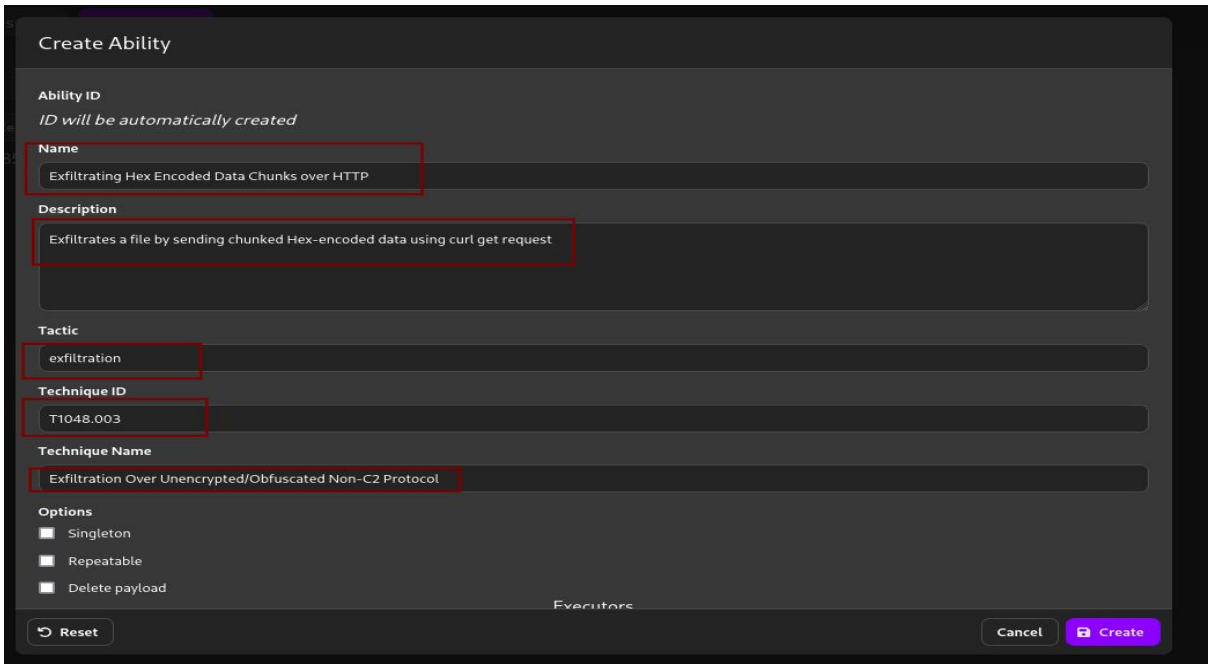



*Figure 3.10 Shows creating a new ability*

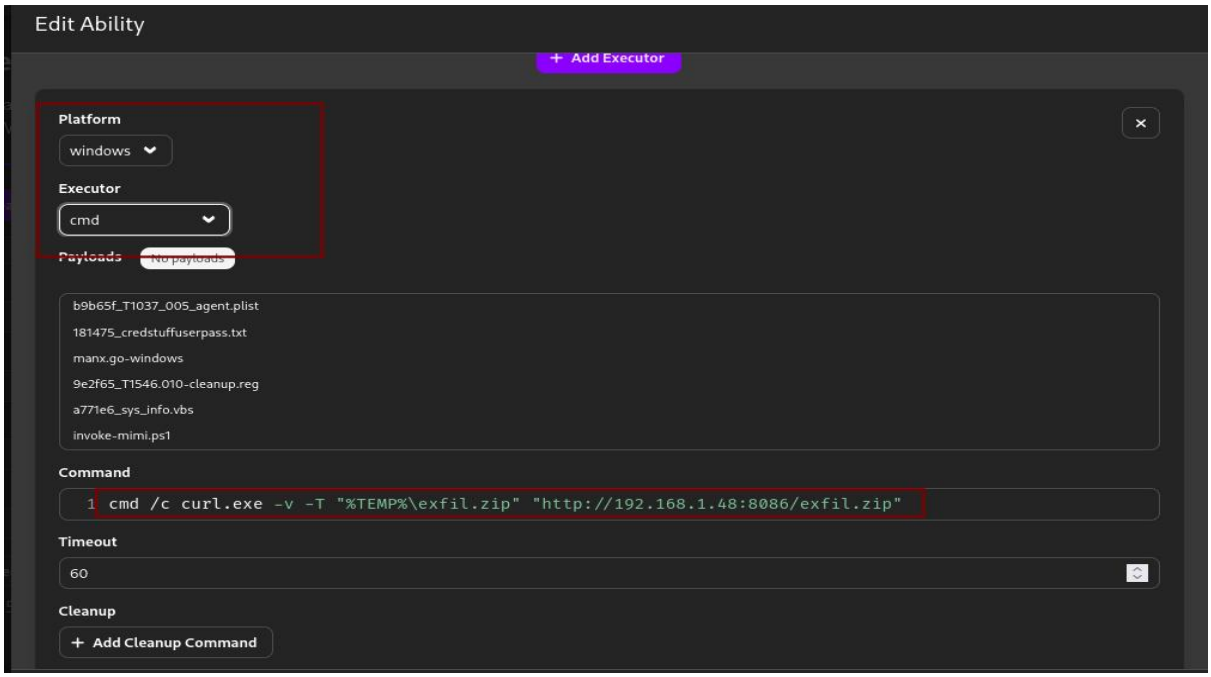



*Figure 3.11 Shows making changes in executor in the new ability*

***Step 11:*** Make a separate Python web-server to receive the ex-filtrated data from the Victim (Windows 10)

~/put_server.py - Mousepad

File Edit Search View Document Help

```python
from http.server import SimpleHTTPRequestHandler, HTTPServer

class CustomHandler(SimpleHTTPRequestHandler):
    def do_PUT(self):
        path = self.translate_path(self.path)
        length = int(self.headers['Content-Length'])
        with open(path, 'wb') as output_file:
            output_file.write(self.rfile.read(length))
        self.send_response(201, "Created")
        self.end_headers()

server_address = ('0.0.0.0', 8086)  # Change port if needed
httpd = HTTPServer(server_address, CustomHandler)
print("Listening for incoming files on port 8086...")
httpd.serve_forever()
```

*Figure 3.12 Shows python script for catching exfiltratig data*

```
┌──(kali㉿kali)-[~]
└─$ python3 put_server.py
Listening for incoming files on port 8086...
192.168.1.46 - - [12/Sep/2025 23:46:10] "PUT /exfil.zip HTTP/1.1" 201 -
192.168.1.48 - - [12/Sep/2025 23:47:02] "GET / HTTP/1.1" 200 -
192.168.1.46 - - [12/Sep/2025 23:53:44] "PUT /exfil.zip HTTP/1.1" 201 -
```

*Figure 3.13 Shows python script running*

***Step 12:*** Creating a ***Custom Adversary Profile***

Now that we have prepared all the abilities, the next step is to create a new adversary profile. Navigate back to the adversaries tab and click New Profile.

The list of the abilities that we are going to add to the Adversary Profile.

- Download Macro-Enabled Phishing Attachment
- Create a Process using WMI Query and an Encoded Command
- Winlogon HKLM Shell Key Persistence – PowerShell
- Identify local users
- Zip a Folder with PowerShell for Staging in Temp
- Exfiltrating Hex-Encoded Data Chunks over HTTP

save this before ***operation phase***

After saving the profiles, it looks like the one displayed below.

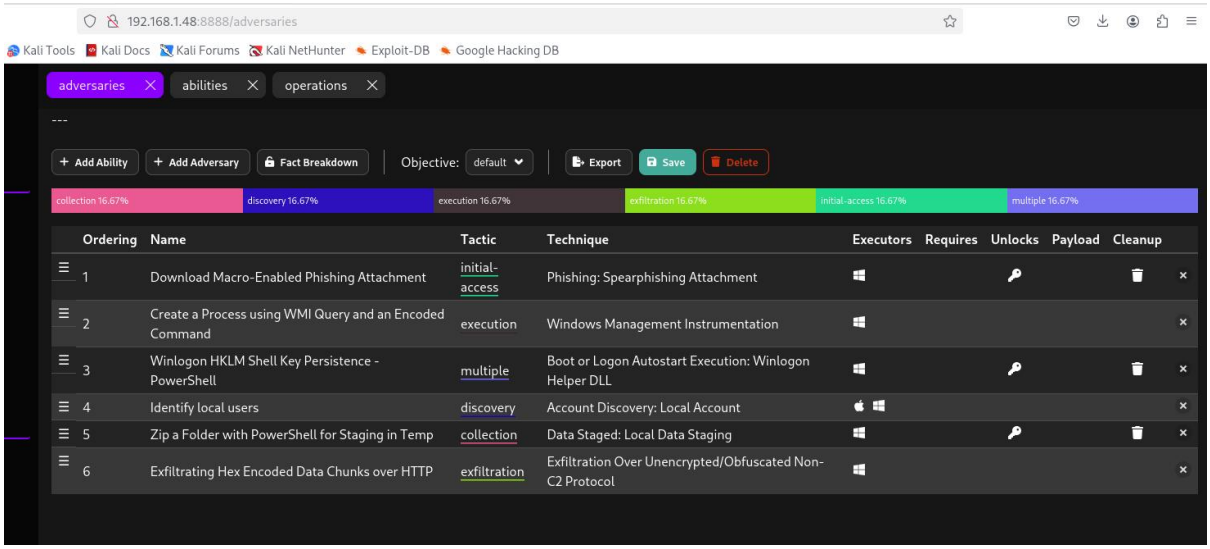
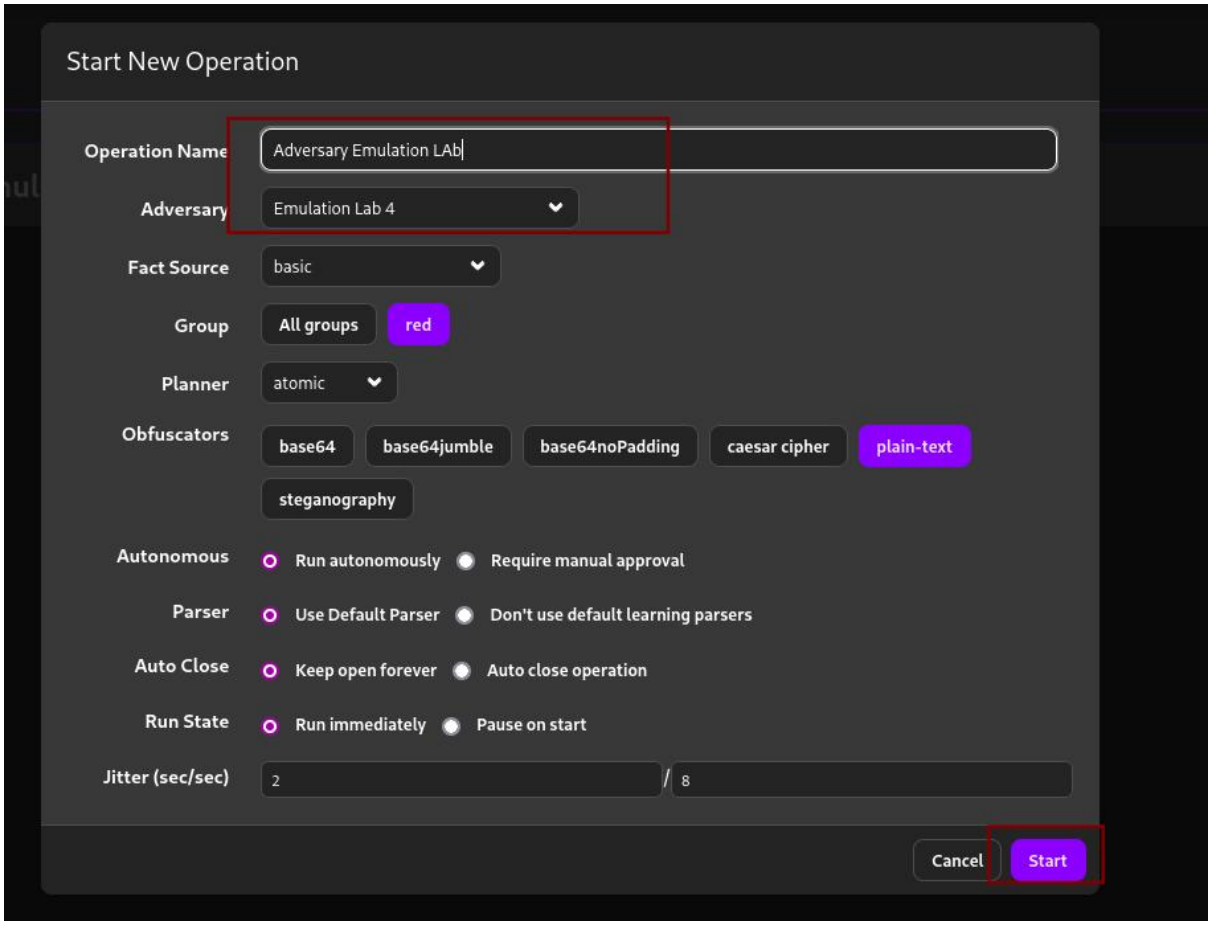



*Figure 3.14 Shows adversary phases*

***Step 13:*** Running the Operation ,select the lab name we kept in above phase and add to the operations.After all the process successfully runs we get the following ,




| Time Ran | Status | Ability Name | Tactic | Agent | Host | pid | Link Command | Link Output |
|---|---|---|---|---|---|---|---|---|
| 9/12/2025, 11:49:36 PM EDT | success | Download Macro-Enabled Phishing Attachment | initial-access | ezvkta | DESKTOP-VT1A6VA | 9784 | View Command | No output |
| 9/12/2025, 11:49:51 PM EDT | success | Create a Process using WMI Query and an Encoded Command | execution | ezvkta | DESKTOP-VT1A6VA | 7368 | View Command | View Output |
| 9/12/2025, 11:50:21 PM EDT | success | Winlogon HKLM Shell Key Persistence - PowerShell | multiple | ezvkta | DESKTOP-VT1A6VA | 9044 | View Command | No output |
| 9/12/2025, 11:51:16 PM EDT | success | Identify local users | discovery | ezvkta | DESKTOP-VT1A6VA | 5880 | View Command | View Output |
| 9/12/2025, 11:51:56 PM EDT | success | Zip a Folder with PowerShell for Staging in Temp | collection | ezvkta | DESKTOP-VT1A6VA | 8312 | View Command | No output |
| 9/12/2025, 11:52:51 PM EDT | success | Exfiltrating Hex Encoded Data Chunks over HTTP | exfiltration | ezvkta | DESKTOP-VT1A6VA | 4104 | View Command | View Output |

*Figure 3.15 Shows operation phase successfully created and executed*

***Step 14:*** Exfiltrated file received in the Webserver.

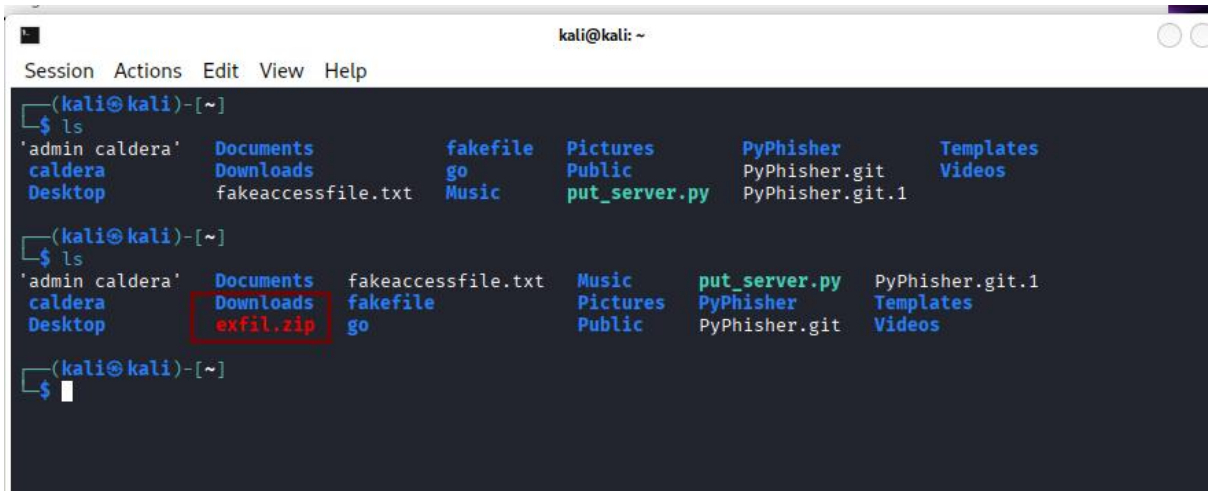



*Figure 3.16 Shows exfil data successfully received on attacker machine*

***Step 15:*** Once all the operations are run successfully , go to temp folder find event logs ,here all the caldera logs are saved.


*Figure 3.17 Shows caldera logs*

## 5. Logging

| *Phase* | *TTP* | *Tool Used* | *Notes* |
|---|---|---|---|
| Phishing | T1566.001 | PyPhisher | Credential harvest |
| Delivery | T1204 | Metasploit | User-executed payload |
| Execution | T1059 | Metasploit | Reverse shell established |
| Exfiltration | T1048.003 | Caldera | Ex filtrated data in hex-encoded chunks |

*Table 5.1 Shows log table for different phases*

## 6. Summary

This lab emulated an APT29-style phishing attack using Py-phisher, Metasploit, and Caldera to test blue-team detection and persistence. Activities included credential harvesting (T1566.001), payload delivery, persistence and exfiltration.