WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Enhancing Security Protocols: An Analysis of Linux Root Password

[1]Rujuta S. Barve

Computer Engineering
[1]rujutabarve20@gmail.com

ABSTRACT—

This report presents a focused analysis of vulnerabilities surrounding Linux root passwords and proposes effective strategies to enhance security protocols. Root accounts, holding ultimate administrative privileges, are prime targets for attackers. Vulnerabilities such as weak passwords, password reuse, insider threats, social engineering, and default passwords expose systems to exploitation. To mitigate these risks, robust defences are necessary. Recommendations include implementing strong password policies, two-factor authentication, regular password rotation, adherence to the least privilege principle, monitoring and auditing, and user education. By addressing these vulnerabilities comprehensively, organizations can strengthen their Linux security posture and reduce the likelihood of unauthorized access and data breaches

Keywords— **Security, Network, Password, Linux, Attack**

Introduction

In the ever-evolving landscape of cybersecurity, the protection of sensitive information and critical systems is paramount. Among the foundational elements of secure computing environments lies the management of root passwords on Linux systems. Root accounts wield unparalleled administrative privileges, making them prime targets for malicious actors seeking unauthorized access. However, despite their pivotal role, root password security often remains an overlooked aspect of overall system defence strategies. This report embarks on an exploration of Linux root password vulnerabilities and proposes effective defence mechanisms to fortify organizational security postures. By delving into the inherent weaknesses of root password management and elucidating robust defence strategies, this analysis aims to equip organizations with the knowledge and tools necessary to safeguard their Linux infrastructure against potential breaches. Through a proactive and comprehensive approach to security, organizations can mitigate risks and bolster resilience in the face of evolving cyber threats.

Body Of The Paper

Linux operating systems are widely utilized across various domains, ranging from enterprise servers to embedded devices. One of the fundamental aspects of securing Linux systems is safeguarding the root account, which holds unparalleled administrative privileges. However, root password security often faces challenges, leaving systems vulnerable to exploitation by malicious actors. This analysis delves into the vulnerabilities associated with Linux root passwords and proposes robust defence mechanisms to mitigate these risks effectively.

Overview

Root passwords hold ultimate administrative privileges on Linux systems, making them prime targets for attackers. Protecting these passwords is essential for maintaining the integrity and security of the entire system.

A. Identified Vulnerabilities:

- Weak Passwords: Easily guessable or commonly used passwords increase the susceptibility to brute-force attacks.
- Password Reuse: Reusing root passwords across multiple systems amplifies the impact of breaches.
- Insider Threats: Malicious insiders with access to root passwords pose significant risks.
- Social Engineering: Manipulation tactics to obtain root passwords through deception.
- Default Passwords: Failure to change default root passwords exposes systems to exploitation.

B. Defensive Strategies:

- Implementing Strong Password Policies: Enforcing complex and unique password requirements.
- Two-Factor Authentication: Adding an extra layer of security beyond passwords.
- Regular Password Rotation: Periodically changing root passwords to mitigate risks.
- Least Privilege Principle: Limiting root access to authorized users only.
- Monitoring and Auditing: Implementing logging mechanisms to track root account usage.

Case Study

1] Capital One Data Breach-2019

Capital One was one of the first banks in the world to invest in migrating their on-premise datacentres to a cloud computing environment, which was impacted by the data leak incident in 2019. Amazon lists Capital One migration to their cloud computing services as a renowned case study (AWS, 2018).

2] Ticketmaster-2021

At the very beginning of 2021, Ticketmaster pleaded guilty to a charge of repeatedly and illegally accessing competitors' computers. Ticketmaster employees repeatedly—and illegally—accessed a competitor's computers without authorization using stolen passwords to unlawfully collect business intelligence.

3] Microsoft-2021

Microsoft stated that it had suffered a cyberattack at the hands of Chinese hacking group Hafnium. The attack targeted hundreds of thousands of on-premises servers across United States that were running Microsoft's Exchange email software, and affected local governments and government agencies as well as businesses, exposing the email communications of each affected organization. Hafnium gained access to the on-prem servers in two ways: via an undisclosed Exchange vulnerability, and by using stolen passwords. Once they accessed the servers, Hafnium created web shells around them, emailing them to steal email data remotely.

4] DailyQuiz-2021

In January, quiz website DailyQuiz suffered a breach that gave hackers access to a database of almost 13 million accounts. The attackers stole the plaintext passwords, email addresses and IP addresses of 8.3 million users and put them up for sale on the Dark Web, eventually making its way into the public domain in May having been exchanged through different data brokers.

Thesis Statement

This thesis of "Enhancing Security Protocols: An Analysis of Linux Root Password Vulnerabilities and Defences" aims to examine the vulnerabilities inherent in Linux root passwords and propose comprehensive defence strategies to strengthen security protocols, mitigating the risks associated with unauthorized access and potential breaches in Linux environments.

Password Types That Are Considered

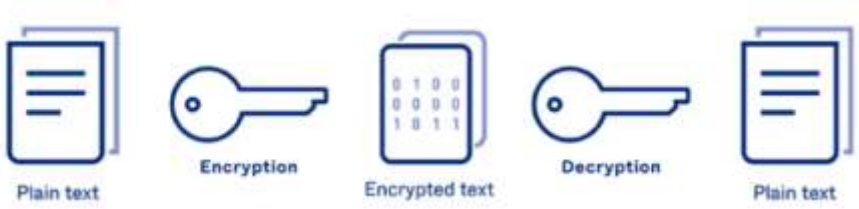
Encouraging users to create complex passwords that include a combination of uppercase and lowercase letters, numbers, and special characters makes them harder to guess or crack through brute-force attacks. These are some ways passwords are stored by the users:

1]Encrypted passwords: An encrypted password is a form of data security where the original password is converted into ciphertext, making it unreadable to anyone without the decrypting the password. This process uses an algorithm that converts the plaintext password into a scrambled form that is a combination of random letters and alphabets. Encryption is crucial for protecting sensitive information, ensuring that even if a data breach occurs and passwords are accessed, they remain secure.

Here as an example,

Original Password: HELLO

Encrypted Password: YUQQW

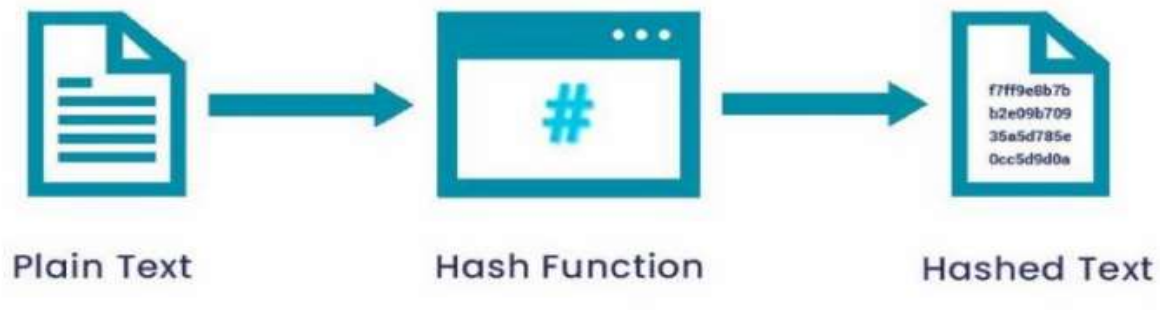


2]Hashed passwords: Hashing is a one-way function. It takes your original password and transforms it into a random string of characters. This process uses a mathematical algorithm. The hashed version is what gets stored on the server, not the actual password. To make the hashing process even more secure, there's seasoning, or more precisely salts and peppers. Salts are random strings of characters that are generated and added to the password before it is hashed. Pepper is a secret value that is added to the password before the hashing process.

Here as an example, if we use the input password - HelloWorld1234, this will be the result:

Original Password: F@rLimit*$!

Hashed password: d85669f719322d44b885c78d1e035ff0bc03e7ab



3]Plain Text Passwords: A plain text password is a password that is entered and saved in a clear, readable format. This type of password is not encrypted and can be easily accessed and read by other humans and machines. When someone stores passwords in plain text, anyone can read them, which is a terrible practice. Storing a password in plaintext may result in a system compromise. Hackers can easily read plaintext passwords and get an access over the system. Hence to avoid any miscellaneous activity or any unwanted access on the system the user is suggested to not to store the password in this format.

Here as an example, we set the password as: password123

In this case, the password will be saved as it is.

Conclusions

The analysis of Linux root password vulnerabilities underscores the critical importance of implementing robust security protocols to safeguard against potential breaches. Weaknesses in root password management, such as the use of weak passwords, password reuse, insider threats, social engineering, and default passwords, pose significant risks to the integrity and confidentiality of Linux systems. However, by adopting proactive defence mechanisms and mitigation strategies, organizations can effectively enhance their security posture. Measures such as implementing strong password policies, deploying two-factor authentication, enforcing regular password rotation, adhering to the least privilege principle, monitoring and auditing root account usage, and providing education and training to users are essential in fortifying Linux environments against unauthorized access.

Acknowledgment

I am grateful to the individuals who provided assistance and constructive input during the course of this study. Their contributions helped refine the concepts and strengthen the arguments presented in this paper. Special thanks to the organizations and funding sources that provided support and resources, enabling the successful completion of this research project.

References

[1] Michael D. Bauer (2001), “Linux Security: Craig Hunt Linux Library”

[2] Glen D. Singh (2016), “Linux Security: Red Hat Certificate of Expertise in Server Hardening (EX413) and LPIC-3 303 (Security) Exams Guide”

[3] Daniel J. Barrett, Richard E. Silverman, and Robert G. Byrnes (2003), “Linux Security Cookbook.”

[4] Herley, Cormac, Paul C. van Oorschot, and Andrew S. Patrick (2009), “Passwords: If we ‘re so smart, why are we still using them?”, Financial Cryptography and Data Security. pg.- 230-234.

[5] Manber, Udi (1996), “A simple scheme to make passwords based on one-way functions much harder to crack”

[6] Yan, Jianxin, Alan Blackwell, Ross Anderson, and Alasdair Gran (2000), “The memorability and security of passwords: some empirical results.”

[7] Kharod, Seema, Nidhi Sharma, and Alok Sharma (2015) “An improved hashing based password security scheme using salting (ICRITO).”

[8] “ACM Transactions on Information and System Security (TISSEC)”

[9] Florêncio, D., and C. Herley (2007), “A Large-Scale Study of Web Password Habits in Proc.”

[10] How to Crack Hashes with Hashcat — a Practical Pentesting Guide (freecodecamp.org)

[11] https://www.internetsafetystatistics.com/passwordencryption/

[12] AWS. (2018). "How to Cloud" with Capital One. Retrieved from AWS: https://aws.amazon.com/pt/solutions/case-studies/capital-one-enterprise