INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Enhancing Cloud Data Security using Artificial Neural Networks for Users' Account Hijacking Security Threats

Sumeet Gill[1], **Renu Devi**[2]*

1 Professor, Department of Mathematics, M.D. University, Rohtak, India
2 Research Scholar, Department of Mathematics, M.D. University, Rohtak, India

OPEN ACCESS

Received: 20-07-2024
Accepted: 13-08-2024
Published: 27-08-2024

Citation: Gill S, Devi R (2024) Enhancing Cloud Data Security using Artificial Neural Networks for Users' Account Hijacking Security Threats. Indian Journal of Science and Technology 17(34): 3538-3552. https://doi.org/10.17485/IJST/v17i34.2339

* **Corresponding author**.

renudevi.rs.maths@mdurohtak.ac.in

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: To ensure the security of passwords of cloud users' accounts that cannot be decrypted easily by any software or hackers. **Methods:** In this manuscript, we have designed an experimental setup using a feed-forward back-propagation algorithm of Artificial Neural Networks techniques to ensure cloud data security. For this purpose, we have utilized password-based datasets created by us. 70% of the datasets are allocated for training and 30% for testing and validation purposes. In this training, **TRAINLM** training function, **LEARNGDM** adaptive function, performance function is **MSE,** and **PURELIN** and **TANSIG** transfer function to transfer the neurons from input to hidden layer and hidden to the output layer has been used. The minimum learning rate is set to 0.001. **Findings:** A username and password are required to access cloud services. A cloud-based storage system is used to store login credentials. Due to inadequate security, attackers use hacking techniques to gain access to users' accounts. Attackers steal data, resources, or services from these accounts. To ensure the security of passwords of cloud users' accounts, we have designed an experimental setup using a feed-forward back-propagation algorithm of neural network techniques. Through this approach, the passwords are stored in a weight matrix, which is a multi-dimensional structure. **Novelty:** The dimensions of the weight matrix are obscured, making it impossible for the hackers to determine its structure. As a result, we conclude that cloud data is effectively secured on the cloud server.

Keywords: Artificial Neural Networks; Cloud Computing; Data Security; Feed-Forward Back-Propagation Algorithm; Machine Learning

1 Introduction

Cloud computing has become one of the best technology to access the resources, and services that are available over the internet. Cloud computing is becoming a major platform for managing resources, including infrastructure, applications, and management software[(1)]. "Microsoft Azure", "Amazon Web Services" (AWS), and "Google" Cloud Platform are prominent examples of cloud computing services that provide virtual machines, databases, analytics, and storage options. While cloud

computing offers many advantages, it also poses some risks in terms of confidentiality, data protection, and security of information. Existing prevention techniques are not sufficient for preventing unauthorized access by internal personnel[2–4]. Organizations relying on cloud computing services experience financial setbacks as a result of data breaches caused by such unauthorized access. There is a higher risk of data loss and access to sensitive information due to cyber-attacks. Cloud computing has been using encryption algorithms to protect sensitive data during transmission and storage however, these algorithms have not been sufficient to maintain comprehensive data security. The security of data includes several important elements, such as preventing unauthorized access, ensuring data integrity, and ensuring data availability. Some security threats related to unauthorized access via password in the cloud are inadequate authentication and authorization, data breaches, insufficient network security, insecure application programming interfaces, account hijacking, insider threats, etc.

1.1 Unauthorized Access of Cloud User's Account via Passwords

Unauthorized access refers to a situation in which someone is able to access the data, application, system, or software without the permission of the authorized user. It is a common method for unauthorized individuals to hack into the accounts of users by guessing their passwords. By either guessing the passwords manually or utilizing the software, the passwords can be cracked. There are many ways in which unauthorized users can access information regarding their user names, passwords, and personal identification numbers (PINs). Cloud data and services are vulnerable to unauthorized access via passwords. Modern business infrastructure includes cloud computing, which requires strong access credentials - mainly passwords. Malicious actors can gain access to a company via weak or compromised passwords, resulting in data breaches, financial loss, and reputation damage. Even though cloud security measures have evolved over the past decade, passwords remain a popular authentication method, making them the prime target of cybercriminals who wish to steal data. Various techniques are employed by hackers to exploit weak passwords, such as brute force attacks and dictionary attacks. Particularly when users use simple, easy-to-guess passwords or reuse them across multiple accounts, these methods can be alarmingly effective.

1.1.1 Types of Unauthorized Access

- **Brute Force Attacks:**

It is a common practice for automated tools to attempt various combinations of passwords in order to find the correct one. The use of weak passwords facilitates this process and increases its chances of success.

- **Dictionary Attacks:**

In order to gain access, attackers use a pre-compiled list of commonly used passwords (e.g., name@123, 1234567890) to attempt to gain access to the system. It is more likely that this type of attack can be carried out with weak passwords that are common or simple.

- **Phishing:**

Although phishing is capable of targeting any password strength, weak passwords are more likely to be fooled into entering them on fake login pages, due to the fact that they are less likely to use multifactor authentication or other security measures that are used to protect their accounts.

- **Password Spraying:**

During this attack, attackers try a few commonly used passwords against a large number of different accounts at the same time. Passwords that are weak and widely used can make it easier for hackers to gain unauthorized access if the password is weak.

- **Social Engineering:**

Passwords are revealed by attackers by manipulating individuals. The use of weak passwords, especially those that can be easily guessed from personal information, makes them more susceptible to social engineering attacks.

In order to minimize these risks, it is essential that you implement a secure mechanism to keep the password securely stored in the cloud.

1.2 Implications and Consequences of Unauthorized Access via Passwords

Unauthorized access to cloud environments can lead to serious consequences, involving numerous security risks, including data breaches that can expose or steal sensitive information. The business plan, financial information, intellectual property, and personal data are all at risk. Beyond immediate financial losses, there are long-term effects such as a breach of trust and damage to brand reputation. Additionally, increasingly stringent global data protection laws may raise compliance issues. Here are some implications and consequences:

1.2.1 Data Breaches:

Exposure of Sensitive Information: A breach of this kind can result in the exposure of sensitive data such as personally identifiable information (PII), financial records, intellectual property, and confidential information pertaining to a business.

1.2.2 Loss of Confidentiality:

Data that is accessed by unauthorized users compromises privacy and may result in the misuse or sale of private information when it is compromised by unauthorized users.

1.2.3 Financial Costs:

In the event of downtime, customer trust is reduced, and potential business opportunities can be lost, resulting in businesses losing business.

1.2.4 Service Downtime:

When the cloud services are not authorized, a disruption can occur, causing business operations to be disrupted and productivity to be lost.

1.2.5 Compromised Systems:

If the account has been compromised, the attacker may be able to alter or damage data, inject malware, or use the compromised account to infiltrate your network further.

1.2.6 Loss of Customer Trust:

If an organization fails to protect customers' data properly, it is likely to lead to churn among customers and difficulty in attracting new ones in the future.

1.2.7 Market Position:

An organization's competitiveness in the market can be undermined when it suffers repeated breaches of security.

1.2.8 Employee Morale:

An organization's security practices may be compromised as a result of a data breach and this can negatively impact employee morale and trust.

1.3 User's Account Hijacking Security Threats

Typically, hackers would try to exploit this type of security breach by accessing the security credentials of the account and then monitoring the activities of the users' accounts. Data may be altered, false information inserted, and clients routed to fraudulent websites. This type of vulnerability is especially dangerous since hackers can utilize the trust and reliability that users have built up to manipulate the clients.

1.4 Way of Hijacking

During account hijacking, a hacker impersonates the account holder via a compromised email account. The most common methods of account hijacking are phishing, spoofing, guessing passwords, or a combination of these methods. It is common for email accounts to be connected to a user's other online accounts, such as social networks, personal finance accounts, and more. Through the use of the account, the hacker can obtain the personal data of the account owner, conduct financial transactions, create new accounts, request funds from the account owner's connections, and ask for assistance in carrying out an illegal activity.

1.5 Effects on business

If the attackers can take advantage of the information in the cloud account, it can be particularly devastating for an enterprise. If this security threat has happened, the integrity and reputation of the company can be damaged, and private information can be leaked, leading to substantial losses to the business or its clients.

In this research work, our first step is to study the algorithms that are used to store passwords in different cloud storage platforms. AES 256 cryptographic algorithm is commonly used to store the password in the cloud. An overview of the various operating systems and techniques utilized for data storage in various clouds is provided in Table 1.

Table 1. Overview of Various Operating Systems and Techniques that are used in Password Storage in the Cloud

Clouds Name	Operating Systems Name	Algorithm used for data storage
Amazon Web Service	Linux	AES-256
Microsoft Azure	Linux	AES-256
Salesforce	Linux	AES-256
Oracle	Linux	AES-256
Alibaba	Linux/Windows/Android	AES-256

All cloud service providers use the AES 256 cryptographic approach to store passwords in the cloud. This algorithm undergoes the following steps to store passwords in cloud storage.

- **Password Hashing:** Secure hashing algorithms like bcrypt or Argon2 are commonly used to hash passwords before encryption. When the password is hashed, it is transformed into a fixed string of characters in a fixed length.
- **Encryption:** The AES-256 algorithm is used to encrypt the hashed password along with a key of 256 bits to do the encryption.
- **Key Generation:** To generate the key, there are a variety of techniques that are used, such as using a secure random number generator.
- **Key Derivation:** As a result of the password of the user being used, a key derivation function (KDF), such as PBKDF2 (Password-Based Key Derivation Function 2) or bcrypt is employed to calculate the actual encryption key. The KDF (machine learning algorithm) uses user passwords, random data, and an iteration count as input.
- **Encryption with AES 256**: CBC (Cipher Block Chaining) mode encrypts the hashed password using the derived encryption key. This cipher text represents the encrypted password that is generated in the previous step.

The following steps are taken to decrypt the password:

- **Key Derivation:** An encryption key is generated using the same key derivation function, random data, and iteration count used for encryption.
- **Decryption with AES 256:** Deriving the encryption key and decrypting the cipher text, using the AES 256 algorithm in the same mode that is used during the encryption process (e.g., CBC), is the same process as generating it. Hashed passwords are obtained in plaintext.
- **Password Verification:** During the decryption process, the decrypted hashed password is compared to the password that is stored in the database. Passwords that match are considered valid if they match.

The methods mentioned above give a summary of how AES 256 is applied to cloud password storage [5]. The use of cloud servers to store information provides a secure and effective method of storing the information of an organization or individual [6–8]. Despite the utilization of established algorithms, users' account continues to face attacks.

Many researchers are working on cloud account hijacking security threats. In this paper, we have outlined some research that has already been conducted on cloud account hijacking security threats.

In the paper [9], the authors have identified some security issues, particularly hacks of user accounts and services. In addition, the article discusses not only the basics but also some proactive steps that may be implemented to prevent the theft of user information and the loss of services, as well as how to minimize the risk.

In the paper [10] the author provides an in-depth analysis of what happened as a result of account hijacking on The New York Times online in a case of cloud security. There are also detailed incident prevention plans which are presented as well as methods to prevent future incidents of this kind from occurring.

1.6 The Need for Advanced Solutions

A more integrated, advanced algorithm is needed due to the existing algorithm's limitations. Password security and overall cloud security posture can be enhanced through emerging technologies such as Artificial Intelligence. These technologies improve threat detection and automate response processes, thereby reducing the ever-increasing risks associated with cloud computing.

2 Methodology

2.1 Artificial Neural Networks for Cloud Data Security

The architecture and operation of Artificial Neural Networks (ANNs) are computational models that are based on biological neural networks seen in the human brain. The structure of these networks consists of interconnected nodes, called neurons, which are organized into layers [11]. Artificial Neural Networks (ANN) is one of the basic components of deep learning, a type of machine learning focused on the representation of data.

In order to ensure the security and protection of sensitive information, advanced methods for securing passwords against potential breaches are essential to enhance security. Among the innovative approaches that can be used to achieve this goal would be the use of Artificial Neural Networks (ANNs), and specifically the use of the feed-forward back-propagation algorithm. Exploitation of this algorithm can significantly enhance cloud data security, especially when it comes to password management, particularly on a cloud platform [12]. With this approach, passwords are not simply stored in hashed form, but are also processed and protected by a sophisticated neural network model, which continually adapts and learns based on the patterns of access and potential threats. Multi-layered security is provided by this method [13]. A significant advancement in cloud data security can be achieved through the use of the feed-forward back-propagation algorithm of Artificial Neural Networks that is employed for storing and managing passwords in cloud environments [13,14]. Through this method password is encrypted as the weight matrix form and it is difficult to find the dimension of the weight matrix. This encrypted password is impossible to crack for any hacker.

2.2 Experimental Setup

During the experiment, a supervised learning model, the Back-Propagation Network, is used as the training model. The training pattern used in this case is passwords. Password contains seven alphanumeric values and one special character. The characters are converted into binary numbers of eight bits each. This experiment is implemented with the neural network toolbox of MATLAB. Usually, the users' password is save **/etc/passwd** file format in cloud. Most of the time unauthorized persons hack the passwords through hacking techniques such as brute force attacks dictionary attacks, etc. To increase the strength of passwords, we have designed this experimental setup.

In Artificial Neural Networks, TRAINLM is used for training feed-forward back-propagation algorithms because of its high accuracy, adaptability, robustness, and fast convergence. With these characteristics, it is the ideal training solution for applications that require efficient and reliable neural network training, like enhancing cloud data security.

For this training, firstly, we have taken the password which is 128 bits. We have proposed the Feed-Forward Back-Propagation Algorithm of Artificial Neural Networks with 60 neurons and LEARNGDM adaptive function, performance function is MSE and PURELIN and TANSIG transfer function to transfer the neurons from the input to the hidden layer and hidden to the output layer. The network has a minimum error of 0.001 and initial weights are zero to one. Figure 1 represents the flowchart which shows how the passwords are memorized by the system using Artificial Neural Networks.

We have trained the two passwords which are 128 bits of data after converting them into the binary form using the same parameters which are used in the above training. 70% of the data is used the training, 15% data is used for test, and 15% data is used for the validation of the data. Table 2 shows the training input, target, and results of this training, Table 3 represents the test input, target, and results and Table 4 describes the validation input, target, and results. Based on our findings, the model performs well for most training instances when the input matches the target (e.g., (1,1, 1,1)), and the results are very close to 1.

Table 2. Training Input, Target, and Results

Training Input		Training Target		Training Results	
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.000000002224220	0.000000005678190
0	0	0	0	−0.000000002224220	0.000000005678190

Continued on next page

Table 2 continued

0	0	0	0	−0.000000002224220	0.000000005678190
1	1	1	1	0.999999995122243	1.00000001203126
1	0	1	0	1.00000000230613	−0.00000000731407
0	0	0	0	−0.000000002224220	0.000000005678190
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.000000002224220	0.000000005678190
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.000000002224220	0.000000005678190
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.000000002224220	0.000000005678190
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
0	1	0	1	0.000000025863348	0.999999945241164
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
1	0	1	0	1.00000000230613	−0.00000000731407
0	0	0	0	−0.000000002224220	0.000000005678190
0	0	0	0	−0.000000002224220	0.000000005678190
1	0	1	0	1.00000000230613	−0.00000000731407
0	0	0	0	−0.000000002224220	0.000000005678190
0	0	0	0	−0.000000002224220	0.000000005678190
1	0	1	0	1.00000000230613	−0.00000000731407
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.000000002224220	0.000000005678190
0	1	0	1	0.000000025863348	0.999999945241164
0	0	0	0	−0.000000002224220	0.000000005678190
1	0	1	0	1.00000000230613	−0.00000000731407
0	1	0	1	0.000000025863348	0.999999945241164
0	0	0	0	−0.000000002224220	0.000000005678190
0	1	0	1	0.000000025863348	0.999999945241164
0	0	0	0	−0.000000002224220	0.000000005678190
0	0	0	0	−0.000000002224220	0.000000005678190
0	0	0	0	−0.000000002224220	0.000000005678190
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.000000002224220	0.000000005678190
0	0	0	0	−0.000000002224220	0.000000005678190
0	1	0	1	0.000000025863348	0.999999945241164
1	1	1	1	0.999999995122243	1.00000001203126

Table 3. Training Input, Target, and Results

Test Input			Test Target		Test Results
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.00000000222422	0.00000000567819
0	1	0	1	0.00000002586334	0.999999945241164
1	0	1	0	1.00000000230613	−0.000000007314078
0	0	0	0	−0.00000000222422	0.000000005678190
0	0	0	0	−0.00000000222422	0.000000005678190
0	0	0	0	−0.00000000222422	0.000000005678190
1	0	1	0	1.00000000230613	−0.00000000731407
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.00000000222422	0.000000005678190

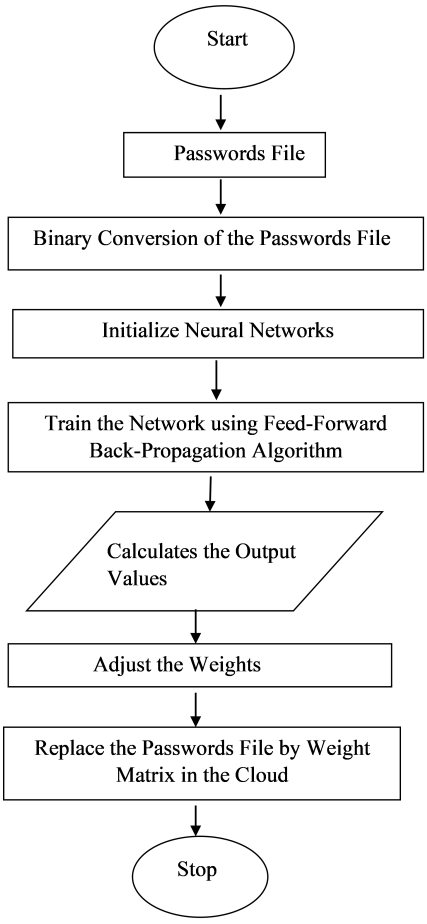

Fig 1. Flowchart of Mechanism to Store Passwords File in the Cloud

Table 4. Validation Input, Target, and Results

Validation Input		Validation Target		Validation Results	
0	0	0	0	−0.00000000222422	0.00000000567819
1	1	1	1	0.999999995122243	1.00000001203126
1	0	1	0	1.00000000230613	−0.00000000731407
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
1	1	1	1	0.999999995122243	1.00000001203126
0	0	0	0	−0.00000000222422	0.000000005678190
1	0	1	0	1.00000000230613	−0.00000000731407
0	1	0	1	0.000000025863348	0.999999945241164
1	1	1	1	0.999999995122243	1.00000001203126

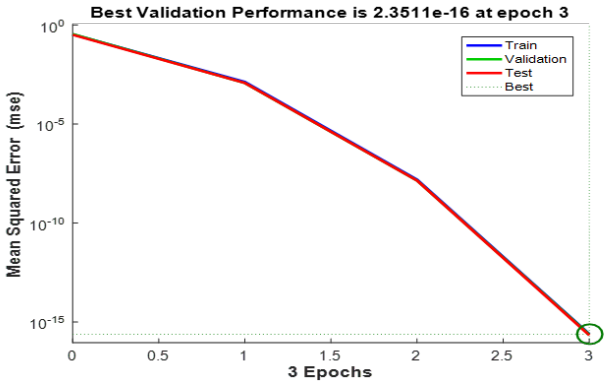

Fig 2. Performance Curve of Training of 128Bits Passwords

Figure 2 shows a decreasing trend in MSE across all three lines (train, validation, and test), indicating improved network performance. Each line converges to the same point, suggesting consistency. According to the results obtained from this specific model and dataset, the best performance (lowest validation MSE) was achieved by the third epoch, which indicates that optimal training duration could be achieved.

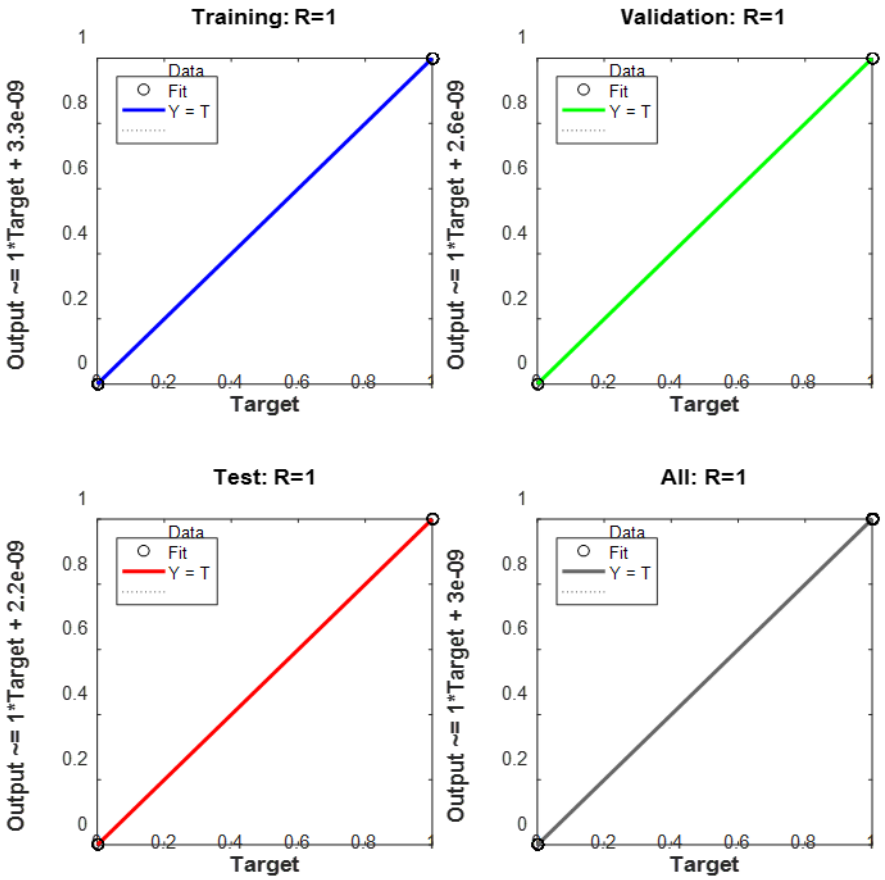

Fig 3. Regression Plot of Training of 128Bits Passwords

Figure 3 shows the regression plots for a neural network's training, validation, and testing phases. In the training, validation, test, and all plot R = 1, the blue line, green line, red line, and black line closely aligns with the dotted line (Y = T), indicating perfect results in this training. Figure 3 shows that all data points, including training and validation, fall on the line, indicating that the model has learned the training dataset well and is generalized very well to both existing and new datasets. Throughout all phases, the data points are aligned with the line consistently, demonstrating the model's robustness and reliability.

After this training, we have analyzed the results, and we can say that machine has memorized the passwords to very close approximation. After that, we recalled new input data using the same weights and bias values which are used in the above training and achieve the appropriate results. Table 5 is represents the input and outputs data which are used for recalling. We use new input data (1, 0) for recalling, corresponding to which we get output values 1.00000000230613 and 1.00000000230613 which are approximate to (1,0).

Table 5. Recalling Input and Results

Recall Input		Recall Outputs	
1	0	1.00000000230613	−0.00000000731407845

Table 6. Input Data for Five Passwords Training

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
…	…	…	…	…	…	…	…
0	0	1	0	1	0	0	1
0	0	0	1	1	0	1	0
1	1	1	1	0	0	0	1

Table 7. Input Data for Ten Passwords Training

0	1	0	1	0	0	1	0
0	1	0	0	0	0	0	1
…	…	…	…	…	…	…	…
0	0	1	1	0	1	1	1
0	0	1	1	1	0	0	0

Table 8. Input Data for 15 Numbers of Passwords Training

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
…	…	…	…	…	…	…	…
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0

Table 9. Input Data for 25 Numbers of Passwords Training

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
…	…	…	…	…	...	…	…
1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0

3 Results and Discussion

The "feed-forward back-propagation algorithm" is used to boost the security of data. In this experiment, initially, we have trained a single password using the feed-forward back-propagation technique. After the one password training, the target and

output values are identical. This indicates that there is no difference between the output and target values. When we train two passwords together, our target and output values are approximately equal which are shows in Tables 2, 3, 4 and 5. Using the same parameters, we've trained 3 passwords, 4 passwords, up to 15 passwords, and 25 numbers of passwords simultaneously. It takes more epochs to train 15 numbers of passwords together. In the training of 25 numbers of passwords together; there is more difference between target and output values. Figures 4, 5, 6, 7 and 8, illustrate the performance of the training of one, five, ten, fifteen, and 25 numbers of passwords with training performance values are 4.5334 e^{-25}, 0.14842, 0.15232, 3.3609 e^{-08}, and 0.17148 respectively.

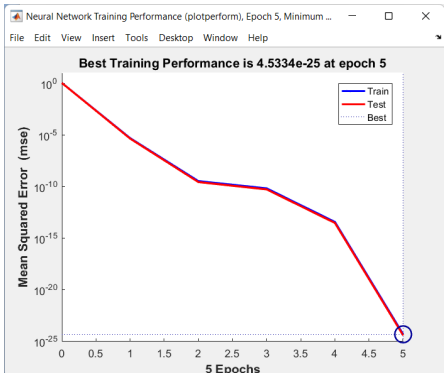

Fig 4. Performance Curve of Single Passwords Training

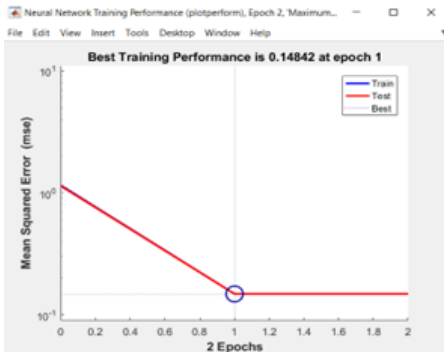

Fig 5. Performance Curve of Five Password Training

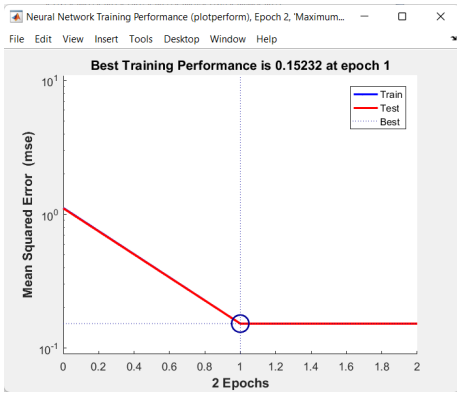

Fig 6. Performance Curve of Ten Passwords Training

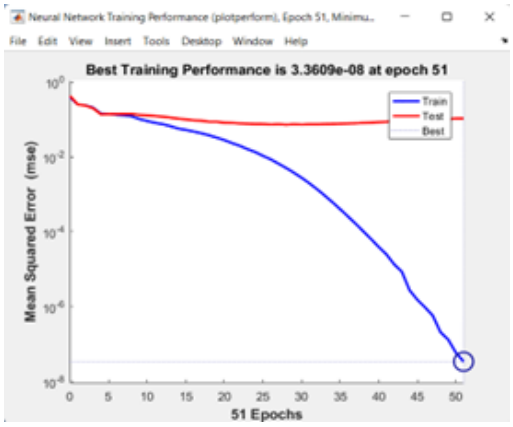

Fig 7. Performance Curve of Fifteen Passwords Training

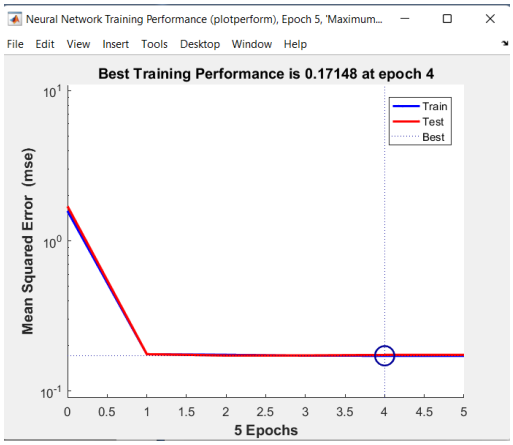

Fig 8. Performance Curve of 25 Numbers of Passwords Training

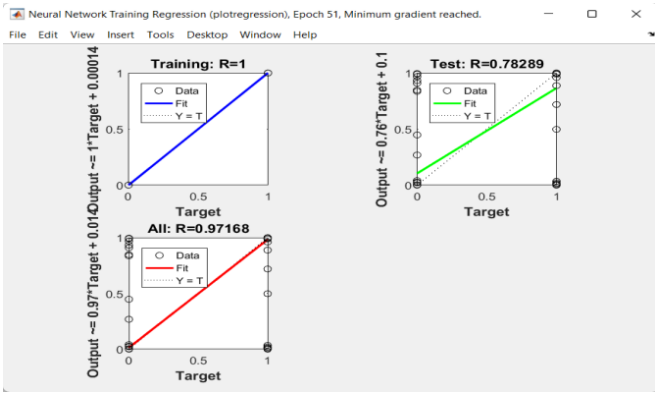

Fig 9. Regression Analysis of Single Password Training

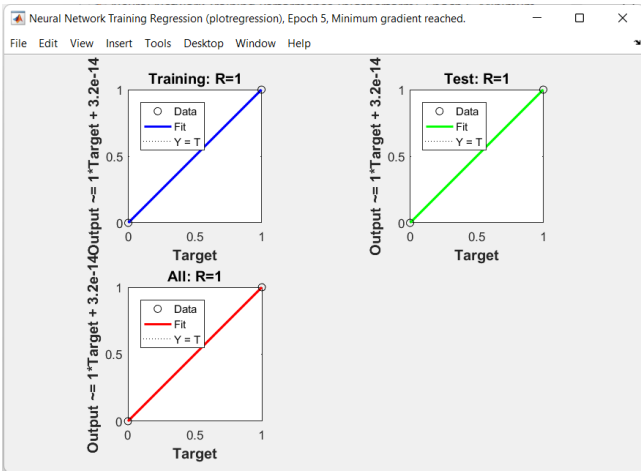

Fig 10. Regression Analysis of Single Password Training

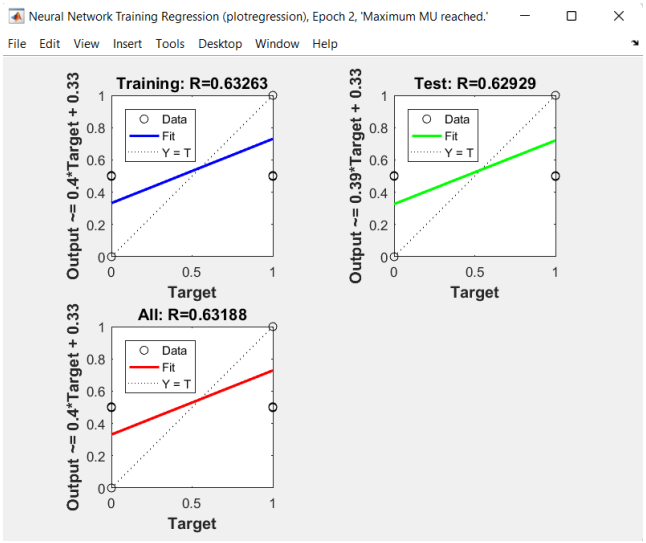

Fig 11. Regression Analysis of Ten PasswordsTraining

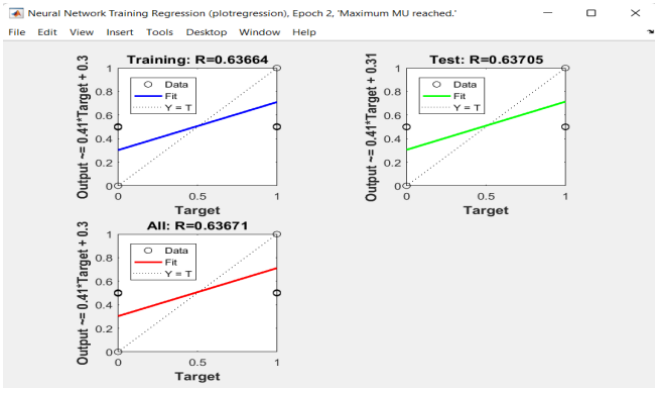

Fig 12. Regression Analysis of Fifteen Passwords Training

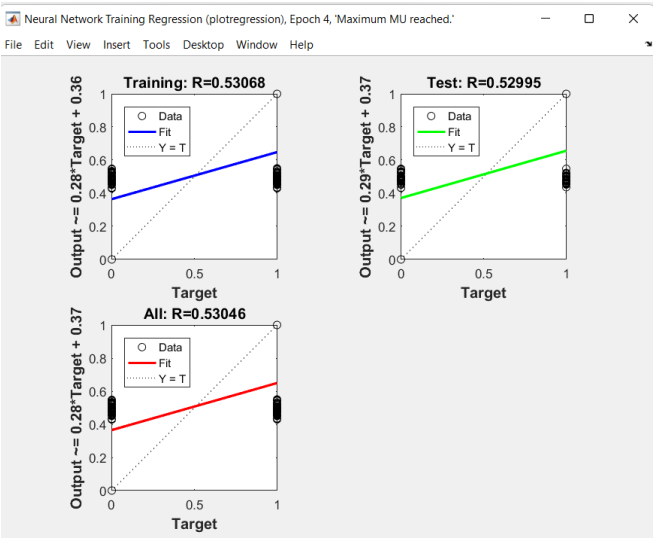

Fig 13. Regression Analysis of Twenty-Five Number of Passwords Training

Figures 9, 10, 11, 12 and 13 are the regression plots of training of one, five, ten, fifteen, and 25 numbers of passwords. These figures include the three graphs: training, test, and validation. In these graphs, R is the correlation between target and output values. When a single password is trained, R has a value of one in the regression analysis, which tells us that this password training is successful. After all the training, we observe that the value of R decreases as the number of passwords increases. Table 10 displays the mean square error (MSE) and the total number of training epochs for different lengths of passwords.

Table 10. Number of Epochs and MSE of the Training of Passwords of Different Lengths

Number of Passwords	M.S.E	Numbers of Epochs
1	0.00002	5
2	0.0623	3
3	0.0889	4
4	0.1170	8
5	0.1484	1
6	0.1522	5
7	0.1522	2
8	0.1565	3
9	0.1514	2
10	0.1523	1
11	0.1120	2
12	0.1075	4
13	0.1670	2
14	0.1090	2
15	0.0094	51
25	0.1797	2

Considering the outcomes of training of one password, two passwords up to fifteen and twenty-five, we have analyzed that, we can store the 14 number of passwords with maximum accuracy and minimum time with minimum error, whereas the 15 number of passwords training has taken more time and more epochs.

Past Research

Two research papers have previously addressed user account hijacking security threats. One paper[9] suggests some theoretical proactive measures to prevent exposure to security threats by hackers, while the other[10] provides strategies to reduce the risk of account hijacking.

In the paper[9] authors suggested the following proactive measures:

- Establishing a protocol that allows employees to share account credentials.
- Track employee usage to make sure there are no unauthorized activities.
- Make sure users and services don't share account credentials in high-confidential places.
- Monitor for unauthorized activity proactively.
- Before signing up for a cloud network, learn about security policies and SLAs.

In paper[10] provides the following strategies to reduce the risk of account hijacking:

- Passwords should be strong, and unique, and multi-factor authentication should be implemented, as well as user access policies.
- Monitor account activity and audit logs regularly to detect unauthorized access, using intrusion detection tools and restricting access to sensitive log information.
- Ensure antivirus software is updated, train users on how to identify threats, and implement secure email practices to fight phishing.

In our current research, we explore a novel approach by using passwords of varying lengths, which are memorized by the machine. The innovation in our work lies in storing these passwords within a weight matrix, a multi-dimensional structure. For instance, a weight matrix of order 1024 can be represented in various forms such as 2×512, $2\times2\times256$, $2\times4\times128$, or $8\times8\times16$, among others. This variability makes it difficult for hackers to determine the exact configuration of the multi-dimensional weight matrix, enhancing the security of password storage in the cloud compared to existing methods.

4 Conclusion

This study has proposed a machine learning-based "feed-forward back-propagation algorithm" of the "Artificial Neural Networks" model that improves data security in the context of users' account hijacking security threats of cloud computing. In this experiment, we have taken 64-bit lengths of passwords then 64×2, 64×3, 64×4, 64×5, 64×6, 64×7, 64×8, 64×9,…. 64×15, 64×25 for the training of the machine. For all the different lengths password training, we get a minimum mean squared error which has shown in Table 10. This simulation protects the users' account from any attackers who conduct financial transactions, create new accounts, and request funds from the account owner's connections. Using this simulation, the password file of the users' account can be replaced with the weight matrix and hackers cannot determine the dimension of the weight matrix by any hacking techniques. In the future, it can be possible to implement this simulation over the medical records to integrate the results. Artificial Intelligence and Neural Networks have a wide range of applications in security, and a large amount of research is possible.

The strength of our research is that passwords are securely stored in the cloud, allowing users to utilize cloud services with confidence in data security. However, a limitation is that cloud service providers struggle to store more than 15 passwords simultaneously within a minimal time and a limited number of epochs. To address this, deep learning algorithms could be employed. Additionally, this research is not suitable for large datasets.

References

1) Kelechi DA. Machine learning algorithms for cloud security. *Advances in Image and Video Processing*. 2023;11(4). Available from: https://doi.org/10.14738/aivp.114.15210.
2) Darbutaitė E, Stefanovič P, Ramanauskaitė S. Machine-learning-based password-strength-estimation approach for passwords of Lithuanian context. *Applied Sciences*. 2023;13(13):13–13. Available from: https://doi.org/10.3390/app13137811.
3) Kumar LR, Ashokkumar C, Pandey PS, Kannaiah SK, Balajee J, Hussan MIT. Security enhancement in surveillance cloud using machine learning techniques. *International Journal of Recent Innovation in Trends in Computing and Communication*. 2023;11(2):46–55. Available from: https://doi.org/10.17762/ijritcc.v11i3s.6154.
4) Awadh WA, Alasady AS, Hashim MS. A multilayer model to enhance data security in cloud computing. *International Journal of Electrical and Computer Engineering*. 2023;32(2):1105–1114. Available from: https://doi.org/10.11591/ijeecs.v32.i2.
5) Soveizi N, Turkmen F, Karastoyanova D. Security and privacy concerns in cloud-based scientific and business workflows: A systematic review. *Future Generation Computer Systems*. 2023;148:184–200. Available from: https://doi.org/10.1016/j.future.2023.05.015.

6) Bamber SS. Cloud secured: A study on cloud computing security. . *Intelligent Systems and Applications in Engineering*. 2023;11:956–966. Available from: https://doi.org/10.1016/j.future.2023.05.015.
7) Sana MU, Li Z, Javaid F, Liaqat HB, Ali MU. Enhanced security in cloud computing using neural network and encryption. *IEEE Access*. 2021;9:145785–145799. Available from: https://doi.org/10.1109/ACCESS.2021.3122938.
8) Rashid SH, Abdullah WD. Enhanced website phishing detection based on the cyber kill chain and cloud computing. *Indonesian Journal of Electrical Engineering and Computer Science*. 2023;32(1):517–529. Available from: https://doi.org/10.11591/ijeecs.v32.i1.
9) Christina AA. Proactive measures on account hijacking in cloud computing network. *Asian Journal of Computer Science and Technology*. 2015;4(2):31–34. Available from: https://doi.org/10.51983/ajcst-2015.4.2.1753.
10) Tirumala SS, Sathu H, Naidu V. Analysis and prevention of account hijacking based incidents in cloud environment. *Proceedings of the 14th International Conference on Information Technology (ICIT)*. 2016;p. 124–129. Available from: https://doi.org/10.1109/ICIT.2015.29.
11) Devi R, Gill S, Narwal E. Securing Account Hijacking Security Threats in Cloud Environment Using Artificial Neural Networks. In: International Conference On Emerging Trends In Expert Applications & Security. Springer Nature. 2023;p. 119–127. Available from: https://doi.org/10.1007/978-981-99-1909-3_11.
12) Odida MO. Exploring the Application of Artificial Neural Networks in Enhancing Security Measures For Cloud Computing: A Survey. *J Mari Scie Res Ocea*. 2024;7(2):1–12. Available from: https://doi.org/10.1007/978-981-99-1909-3_11.
13) Dalal S, Manoharan P, Lilhore UK, Seth B, Alsekait DM, Simaiya S, et al. Extremely boosted neural network for more accurate multi-stage Cyber attack prediction in cloud computing environment. *Journal of Cloud Computing*. 2023;12(1). Available from: https://doi.org/10.1186/s13677-022-00356-9.
14) Zavieh H, Javadpour A, Sangaiah AK. Efficient task scheduling in cloud networks using ANN for green computing. *International Journal of Communication Systems*. 2024;37(5). Available from: https://doi.org/10.1002/dac.5689.