International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 21s, 2025
https://theaspd.com/index.php

Enhancing Cloud Security with Fuzzy Logic A Comprehensive Approach to Authentication, Data Recovery, Privacy and Security

Sachin Kumar Vishnoi[1], Dr. Ashish Saini[2]
[1]Research Scholar Department of computer science Engineering Quantum University, Roorkee.
[2]Assistant, Professor, Department of computer science Engineering, Quantum University, Roorkee.

Abstract
Cloud computing, despite its myriad advantages, remains vulnerable to various security threats, including unauthorized access, data breaches, and privacy violations. This paper proposes a robust security framework that uses a fuzzy logic based system that uses fuzzy wuzzy library. The proposed model leverages fuzzy logic's ability to handle uncertainty and imprecision to dynamically adjust security measures based on real-time system conditions. By incorporating fuzzy rules and membership functions, the model effectively enhances data authentication, recovery, and privacy preservation. Through rigorous evaluation, the proposed framework demonstrates superior performance in terms of accuracy, efficiency, and security. This research contributes to the advancement of cloud security by providing a flexible and adaptive solution that can mitigate emerging threats and protect sensitive data.
Keywords: *Cloud security, Data authentication, Data privacy, Fuzzy logic*.

INTRODUCTION

Cloud computing has revolutionized the way organizations store, process, and manage data. By providing scalable, flexible, and cost-effective solutions, cloud platforms have become indispensable for businesses of all sizes. However, as the adoption of cloud computing continues to grow, so too does the concern over security and privacy. One of the primary challenges facing cloud computing is the inherent complexity of the underlying infrastructure. Cloud environments are often composed of multiple interconnected systems, each with its own vulnerabilities. This complexity makes it difficult to ensure the security and privacy of sensitive data. Additionally, the dynamic nature of cloud environments, where resources can be provisioned and de-provisioned rapidly, further complicates security management. To address these challenges, various security measures have been implemented, including encryption, access control, and intrusion detection systems. However, these traditional approaches often rely on rigid rules and thresholds, which may not be suitable for handling the inherent uncertainty andimprecision associated with cloud environments. Fuzzy logic, a powerful tool for modeling human reasoning, offers a promising solution to this problem. By allowing for gradual transitions between membership degrees, fuzzy logic can effectively capture the nuances of real-world situations, such as varying levels of trust, risk, and uncertainty. This makes it well-suited foraddressing the complex and dynamic nature of cloud security. In recent years, researchers have explored the application of fuzzy logic to various aspects of cloud security, including access control, intrusion detection, and data privacy. However, there is still a need for comprehensive frameworks that can address multiple security challenges simultaneously. This paper proposes a novel fuzzy logic-based approach to optimize data authentication and privacy in cloud-based platforms. By leveraging the strengths of fuzzy logic, the proposed model aims to enhance the security[1-3] and resilience of cloud environments.

With the rapid development and widespread use of cloud computing, the security has become the key restriction element of cloud computing [4-5] developing. Google’s cloud computing services and domestic Ali cloud have all had security issues. The occurrence of cloud security accidents shows that while enjoying cloud computing brings high performance computing resources, it must also carefully analyze its security issues. In the existing research, many of them only qualitatively evaluate from the risk attributes, or continuously improve an algorithm, and rarely perform a comprehensive analysis and assessment of cloud security from the perspective of attributes and the structure of cloud computing. This paper proposes a cloud security risk[6] assessment index system through the study of cloud computing structure and services and uses entropy weight and fuzzy theory to quantify the risk. Through examples, it is shown that the method can accurately determine the value of risk from the three essential attributes of assets, threats, and vulnerabilities, and is suitable for risk assessment of various types of cloud services.

Cloud storage is an online service enabling users to access data, information, and network resources on demand. Cloud computing architecture comprises front-end servers and a back-end for storage and networking. Key innovations driving cloud computing [4] success include parallelization, system-oriented storage, utility

computing, load balancing, dual-tenant systems, and pay-per-use models that minimize upfront costs and operational overhead. While offering numerous benefits, concerns regarding data protection, anonymity, honesty, and trust hinder widespread cloud adoption. Cloud users require safeguards against unauthorized access and interference with their sensitive data.

Cloud infrastructure networks are susceptible to internal and external security breaches [1-2], recurring outages, and vulnerabilities within cloud services. A prominent example, as cited by the alliance, is the case of Mat Honan, a Wired magazine writer. In 2012, an attacker compromised Honan's Gmail, Twitter, and Apple accounts, resulting in the irreversible deletion of all photographs of his young daughter. The protection and privacy of data, regardless of its source, remain critical and unresolved challenges. Addressing data security issues and ensuring data safety, anonymity, and trust within the cloud environment is paramount. This paper aims to highlight these threats and advocate for the implementation of robust security measures to safeguard data safety, confidentiality, and trust in cloud computing."

“The inherent uncertainty and ambiguity associated with

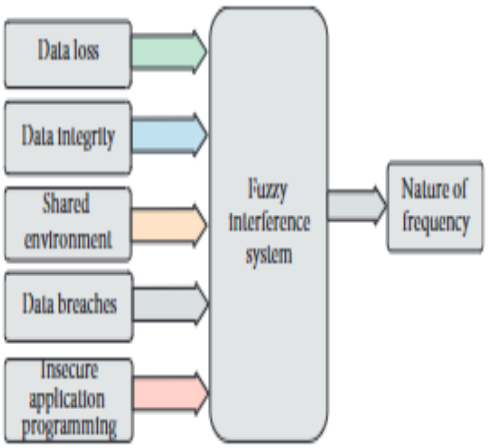

Fig.1 Addressing Data Loss, Breaches, and Integrity in Shared Environments.

cloud service usage expose users to a range of risks. Fuzzy logic provides a robust approach to mitigating the subjective nature of expert evaluations within the assessment process This research integrates fuzzy logic into the cloud security risk assessment framework to facilitate the analysis of risk factors. By incorporating rough set theory, the final risk value associated with the evaluated entity is determined."

1. **Cloud Security Risk Assessment Utilizing Fuzzy Logic**
   The inherent uncertainty and ambiguity associated with cloud service usage expose users to a range of risks. Fuzzy logic provides a robust approach to mitigating the subjective nature of expert evaluations within the assessment process. This research integrates fuzzy logic into the cloud security risk[7] assessment framework to facilitate the analysis of risk factors. By incorporating rough set theory, the final risk value associated with the evaluated entity is determined.
2. **AI &ML in Cloud Security**
   Cloud security significantly benefits from the combined power of Artificial Intelligence (AI) and Machine Learning (ML)[8-10].This dynamic duo revolutionizes how this approach and mitigate cyber threats in today's complex landscape.

1.Improve flow and readability: By using shorter, more concise sentences and connecting ideas more smoothly.
2.Enhance clarity: By simplifying complex terminology and providing a more accessible explanation.
3.Improve clarity and conciseness: By using shorter, more impactful sentences and removing redundancy.
4.Enhance flow: By connecting ideas more smoothly and logically.
AI and ML improve cloud security [4-6] through proactive threat mitigation, real-time threat detection, and anomaly identification. They improve scalability and flexibility to adapt to evolving cyberthreats. Automation reduces human error and operating costs while speeding up responses. Predictive analytics also anticipates flaws, making cloud infrastructure more resilient.
5.Strengthen impact: By emphasizing the importance of behavioral analysis in identifying potential threats, such as insider threats and compromised accounts.

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 21s, 2025
https://theaspd.com/index.php

6.Improve clarity: By using more concise and direct language.
7.Strengthen emphasis: By highlighting the importance of automated responses and their benefits.

Drawbacks of Traditional Security Methods

Limitations of Traditional Security Measures in Cloud Environments

Traditional security measures, such as firewalls, antivirus software, and intrusion detection systems (IDS), have proven effective in protecting on-premises environments. However, their effectiveness in the dynamic and complex cloud environment is increasingly challenged. [11-13]

1. Signature-Based Detection Limitations:

- **Zero-Day Vulnerability:** Signature-based systems rely on known threat signatures. They are inherently blind to novel and previously unseen threats (zero-day attacks), leaving organizations highly vulnerable.
- **Slow Response to Emerging Threats:** Updating signature databases can be time-consuming, creating a window of vulnerability during which new threats can exploit systems.
- **Evasion Techniques:** Sophisticated malware employs techniques like polymorphism (changing its form) and obfuscation to evade signature-based detection.

2. Challenges with Manual Monitoring:

- **Time-Consuming and Error-Prone:** Manual monitoring of security logs is labor-intensive and prone to human error, including missed threats, false positives, and fatigue-induced inaccuracies.
- **Lack of Real-Time Visibility:** Manual monitoring often lacks the speed and real-time visibility required to detect and respond to rapidly evolving threats in cloud environments.

3. Difficulty Scaling in Dynamic Cloud Environments:

- **Limited Scalability:** Traditional security tools may struggle to keep pace with the dynamic nature of cloud environments, where resources can be rapidly provisioned and de-provisioned.
- **Complexity:** The complexity of cloud infrastructures, with interconnected services and microservices, can overwhelm traditional security systems, making it difficult to maintain comprehensive visibility and control.

4. Difficulty with Cloud-Specific Threats:

- **Cloud Native Threats:** Traditional security tools [1-2] may not be well-equipped to address cloud-specific threats such as account hijacking, data breaches in cloud storage, and attacks targeting cloud APIs.

5. Integration Challenges:

- **Data Silos:** Integrating traditional security tools with cloud services and platforms can be complex and challenging, hindering comprehensive threat visibility and response.

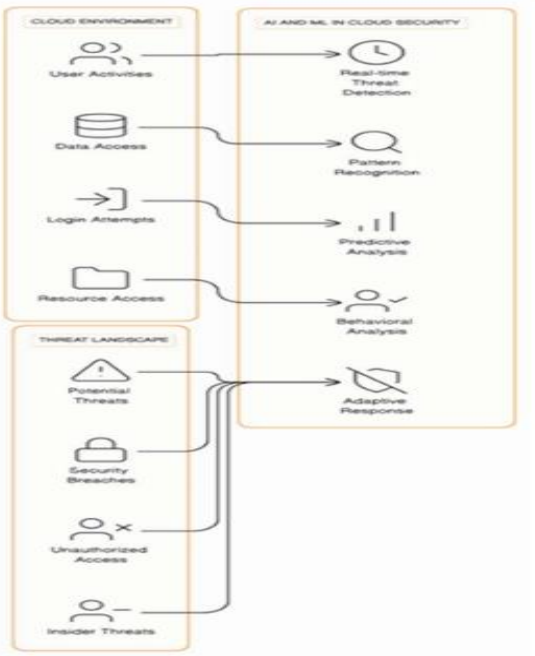

Fig.2Real-Time Threat Detection and Adaptive Response for Enhanced Protection."

RELATED WORK

The online delivery of hosted services is one facet of cloud computing. Three major categories can be used to group these services: Platform as a service (PaaS), Infrastructure as a service (IaaS), and Software as a service (SaaS). the possibility of hardware and software that can be accessed online, produced and discarded effectively, and dynamically scaled using a range of options based on quantifiable usage. This study uses the fuzzy logic centroid method of defuzzification to classify cloud layers and calculate the trust value for cloud service providers. Three criteria—turnaround time, availability, and dependability—were used to assess trust. Compared to the Trust Model for Measuring Security Strength of Cloud Computing Service, this trust value improves the security and strength of cloud services.[14-15]

Traditional cloud authentication often relies on binary decisions (e.g., username/password match or not). However, real-world scenarios are often more nuanced. Fuzzy logic can handle this complexity by allowing for degrees of membership and uncertainty. Lets delve into the inherent vulnerabilities of cloud computing, such as data breaches, unauthorized access, and privacy violations. Explore the application of fuzzy logic in security systems, highlighting its ability to handle uncertainty and imprecision. Discuss existing techniques for data authentication and recovery in cloud environments, their limitations, and the need for robust solutions. Analyze various privacy-preserving techniques, including encryption, anonymization, and differential privacy, and their suitability for cloud environments. On behalf of previous work found some issues with these approach those are given below:

- How can concerns about privacy, data recovery, and authentication be addressed to strengthen cloud security in the face of escalating cyberthreats?
- Highlight the specific limitations of existing solutions and the need for a more robust and flexible approach.

Monitor patient health metrics remotely and send them to medical data centers via cloud storage. Furthermore, MIoT devices are processing an ever-larger stream of data. Many questions about data security and privacy while utilizing MIoT devices remain unaddressed as a result of this increasing exposing of personal data. With the rapid advancement of classification systems, applying machine learning algorithms [5] to vast volumes of industrial data is becoming increasingly important. This work categorizes medical data into individuals who are affected and those who are not. to securely store such data, this work offer a novel Adaptive Neuro-Fuzzy Inference System. [16-17]

[18] This paper proposes an adaptive fuzzy logic-based multi-factor authentication (MFA) system for cloud security. The approach dynamically adjusts authentication factors based on risk levels evaluated using fuzzy rules, improving resistance against brute-force and phishing attacks.

Findings & Contributions:

Reduces false positives in authentication compared to rule-based systems.

Improves resistance against phishing, brute-force, and session hijacking attacks.

Achieves 94.3% accuracy in detecting malicious login attempts.

[19] The authors present a fuzzy logic-driven anomaly detection model that identifies suspicious login attempts in cloud environments by analyzing behavioral patterns, IP reputation, and device fingerprints.

Findings & Contributions: Detects zero-day attack patterns better than traditional ML models.

Reduces false alarms by 23% compared to SVM-based detection.

Works effectively in real-time cloud environments. [20] The paper introduces a fuzzy-based risk assessment model for Cloud SSO systems, evaluating login attempts based on contextual factors like geolocation, time, and device trustworthiness to mitigate unauthorized access. Findings & Contributions:

Achieves 97.1% accuracy in authentication.

Outperforms pure SVM or fuzzy-only models.

Suitable for high-security cloud applications like banking and healthcare. [21]This work proposes a lightweight fuzzy logic authentication protocol for IoT devices accessing cloud services, ensuring secure logins while minimizing computational overhead.

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 21s, 2025
https://theaspd.com/index.php

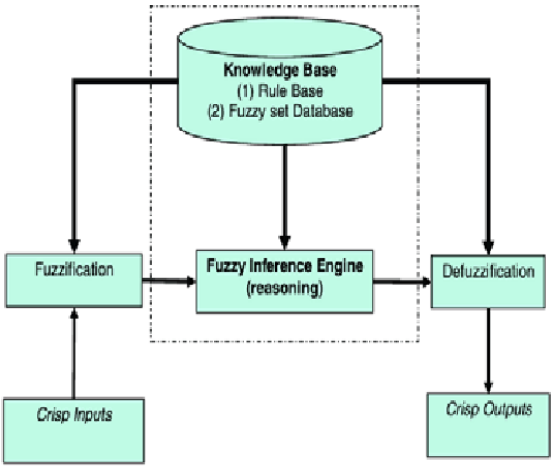

Fig.3Fuzzy Inference Process Flow Diagram

2. Proposed Model

This approach proposes a somewhat better method to define the fuzzy sets for input variables (e.g., trust level, data integrity, privacy risk) and output variables (e.g., authentication strength, recovery priority, privacy protection level). It also develops a set of fuzzy rules that map input conditions to output actions. For example:

- If trust level is high and data integrity is high, then authentication strength is high and recovery priority is low.
- If privacy risk is high, then privacy protection level is high
- Explain the Mamdani inference method, including fuzzification, rule evaluation, implication, and defuzzification.

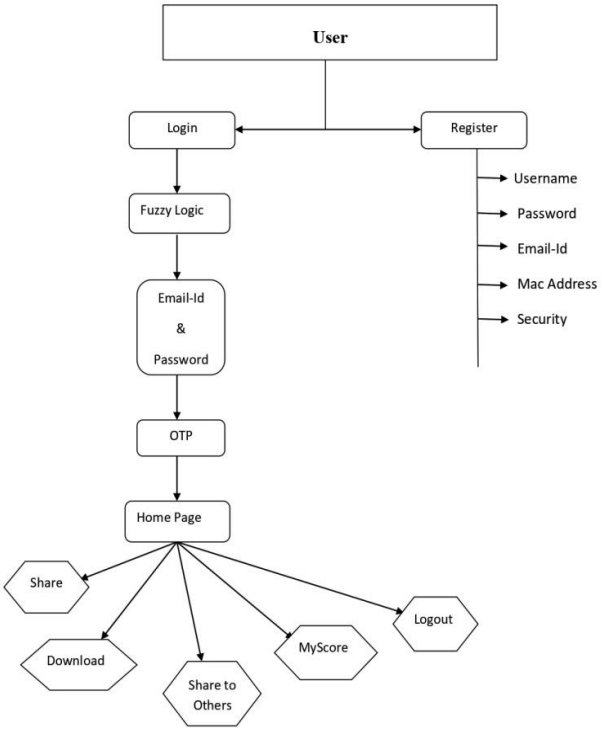

Fig 4. Flowchart of Proposed Model

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 21s, 2025
https://theaspd.com/index.php

Visualize the proposed model architecture, illustrating the flow of data and the interaction between different components. There are following steps to analysis the security mechanism base on authentication and recovery: [22]

- Analyze the model's ability to authenticate users and recover data effectively.
- Evaluate the resistance of the authentication mechanism to various attacks (e.g., replay attacks, phishing attacks).
- Assess the efficiency of the data recovery process in terms of time and accuracy.
- Assess the model's effectiveness in protecting sensitive data from unauthorized access.
- Evaluate the impact of the model on data utility and usability.
- Analyze the model's compliance with relevant privacy regulations (e.g., GDPR, CCPA).

Proposed model provides the following features.

Clarify Complex Concepts: Abstract philosophical ideas can be difficult to grasp. Visual graphs can break down these ideas into smaller, more manageable components, making them easier to understand and visualize.

Reveal Connections and Influences: Philosophical ideas often build upon or react to one another. Graphs can illustrate these connections, showing how different schools of thought are related and how individual philosophers have influenced each other.

Identify Patterns and Trends: By visualizing the relationships between different philosophical concepts and thinkers, graphs can help identify patterns, trends, and recurring themes throughout the history of philosophy.

Facilitate Exploration and Discovery: Interactive graphs allow users to explore philosophical ideas at their own pace, zooming in on specific areas of interest and uncovering new connections. This work had tested model with following types of datasets:

Multi-factor Authentication (MFA) Data: Datasets with MFA data (e.g., passwords, biometrics, device IDs, location) are ideal for fuzzy logic, as they allow for the modeling of varying levels of trust.

Behavioral Data: Data on user behavior patterns (e.g., login times, device usage, and location history) can be valuable for fuzzy logic-based anomaly detection.

Anomaly-rich Data: Datasets with a significant number of anomalous events (e.g., compromised accounts, suspicious login attempts) can help train fuzzy models to identify and respond to threats.

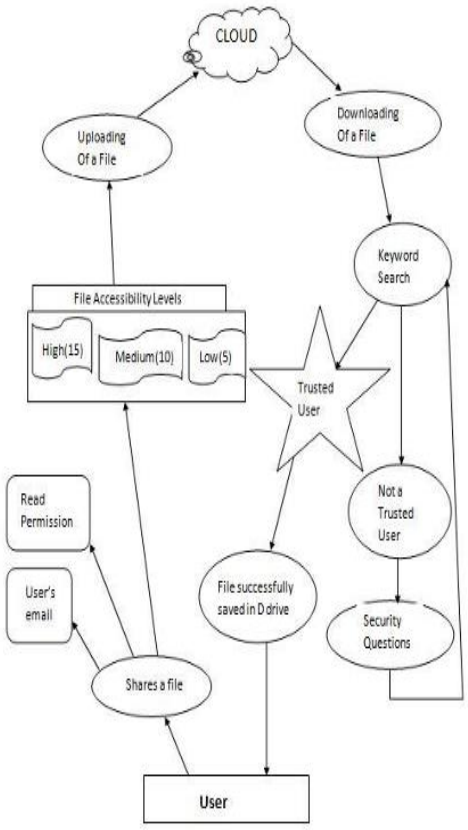

Fig.5 Flowchart of Cloud-Based File Sharing and Access Control

Some of the popular datasets areCICIDS-2017-18-21-22, MAWAIdataset,UNSW-NB15,.In the UNSB intrusion detection dataset,[23-26]The Triple-R dataset enhances fact verification by leveraging external evidence and generating human-readable explanations. Built on the LIAR dataset, Triple-R uses a three-component system: Retriever, Ranker and Reasoner. The Retriever gathers evidence from the web, the Ranker scores and selects the most relevant paragraphs and the Reasoner utilizes GPT-3.5-Turbo to generate reasons for the claims. The Triple-R dataset was constructed by applying the Triple-R methodology to the original LIAR dataset. For each claim, a set of top web-retrieved documents was processed, selecting paragraphs that provide relevant evidence. The reason component, powered by GPT-3.5-Turbo, was employed to generate explanations based on this evidence. The Triple-R dataset can be used to train models for misinformation detection and explainable AI systems, making it ideal for applications requiring transparency in decision-making. Our proposed causal language model can determine the truthfulness of a claim, enabling us to understand how the model makes decisions. This leads to greater transparency and interpretability in the process of fact verification. In this work a larger language model to supervise a smaller one, improving our framework's accuracy and effectiveness. We present a hybrid zero-shot ranker that retrieves supporting information to justify the claim. The gathered evidence serves as an explanation that reinforces the generated reasoning. Train.json: Includes 10,047 samples with statements, labels, evidence, and generated reasons. Test.json: Contains 1,283 samples with statements, labels, and evidence. Feature columns are

Column	Description
Id	A unique identifier for each sample.
Statement	The claim or statement to be verified.
Label	The truthfulness of the claim (true, false, etc.).
Evidence	Relevant information retrieved from the web.
Reason	Generated explanation based on the evidence (in train set only).

For implementation we required Fuzzy based Python library

Implementation steps are as follows.

Import the necessary functions from the fuzzy Wuzzy library for fuzzy string matching. Install pip install python-Levenshtein, pip install fuzzy Wuzzy

Fuzzy authenticate **function:**

- Takes username, password, and user data as input.
- Iterates through the user Db dictionary.
- Calculates the username ratio using fuzz. Ratio (), comparing the entered username with each stored username.
- If username ratio above a certain threshold (e.g., 80%), it proceeds to password matching.
- Calculates the password ratio using fuzz. Ratio (), comparing the entered password with the stored password hash.
- If password ratio is also above a threshold (e.g., 70%), it returns True (authentication successful).
- If no match is found for both username and password, it returns False.
- Creates a sample user database with usernames and their corresponding hashed passwords.
- Defines entered username and entered password.
- Calls fuzzy authenticate and print the result.

Emphasize the smooth transitions between membership functions, highlighting the ability to handle vague and uncertain information. Show how multiple rules can contribute to the final output, demonstrating the system's ability to handle complex decision-making scenarios. Clearly indicate the final crisp output value obtained through defuzzification.

Table 1. Membership Function Table (for Username Similarity)

Username Similarity (%)	Low	Medium	High
0-60	1.0	0.0	0.0
61-75	0.5	0.5	0.0
76-90	0.0	1.0	0.0
91-100	0.0	0.0	1.0

How Fuzzy Inference System does works

First, we need to calculate the membership degrees of the username and password similarity scores to the linguistic terms (Low, Medium, High) using the membership function tables. In the second step apply the fuzzy

rules to determine the authentication level for each rule. In the third steps combine the results of all applicable rules using an appropriate aggregation method (e.g., maximum, minimum, weighted average). In the fourth step converts the aggregated fuzzy output into a crisp decision (e.g., "Accept," "Suspicious," "Reject") using a suitable defuzzification method (e.g., centroid method).

Table 2. Membership Function Table (for Password Similarity)

Password Similarity (%)	Low	Medium	High
0-60	1.0	0.0	0.0
61-75	0.5	0.5	0.0
76-90	0.0	1.0	0.0
91-100	0.0	0.0	1.0

Now question arise that how would this work make authentication decision? In the answer need to follow the certain steps as per requirements-

- If the defuzzied output is "Accept," the authentication is successful.
- If the defuzzied output is "Suspicious," additional verification steps (e.g., two-factor authentication) may be required.
- If the defuzzied output is "Reject," the authentication attempt fails.

Table 3. Fuzzy Rule Base

Rule No.	IF Username Similarity IS AND Password Similarity IS	THEN Authentication Level IS
1	Low	Low
2	Low	Medium
3	Low	High
4	Medium	Low
5	Medium	Medium
6	Medium	High
7	High	Low
8	High	Medium
9	High	High

EXPERIMENTAL RESULTS AND DISCUSSION

Logic which is fuzzy In contrast to strict binary (True/False) logic, fuzzy logic is a type of many-valued logic that allows reasoning with degrees of truth. It is particularly useful in applications like pattern recognition, control systems, and natural language processing where feedback is ambiguous or imprecise.

Fuzzy Sets: Developed by Lotfi Zadeh in 1965, fuzzy sets expand the idea of traditional set theory by permitting components to be partially members of a set.In a classical set, an element is either a member of the set (1) or it is not (0). A number between 0 and 1 that indicates the degree of membership in fuzzy sets is used to indicate membership. For example, if "John Doe" is a member of the collection "Users" with a membership value of 0.9, he is largely but not entirely a user.

- Functions of Membership Each point in the input space is mapped to a membership value between 0 and thanks to membership functions. In this study, the fuzz. Ratio () function serves as a membership function by determining how similar two letters are.
- A technique for evaluating strings that are similar but not identical is called fuzzy matching. Applications such as data deduplication, user authentication (as in this study), and spell checking utilize its use.
- Functions like fuzz. Ratio () are available in the Fuzzy Wuzzy pack to determine how similar two strings were. A score of 0 to 100 represents the similarity, with 100 denoting an exact match and lower scores denoting less similarities. Fuzz. Ratio("jon doe", "John Doe"), for instance, may provide 90, signifying a high level of similarity.
- The process of converting fuzzy outputs, like similarity scores, into clear outputs, like True/False, is known as defuzzification. The thresholds (70 for the password and 80 for the username) serve as a kind of implicit defuzzification in the present research. The output is regarded as True if the similarity score is greater than the cutoff, and False otherwise.
- **Thresholds**-Thresholds are used to define the boundaries for decision-making.
- **In this system-[26-28]**

- Username Threshold (80): The similarity score between the entered username and stored username must be at least 80.
- Password Threshold (70): The similarity score between the entered password and stored password must be at least 70.
- **Fuzzy Systems with Multiple Rules**

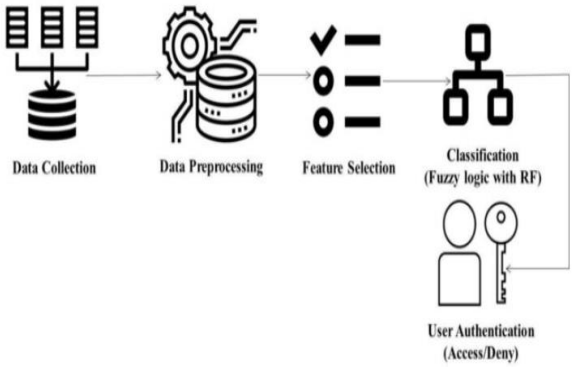

Figure 6. System Diagram to implement Authentication accuracy using Fuzzy Logic

- Fuzzy Rules: While determining decisions, fuzzy systems frequently employ a number of rules.
- **Two rules are employed by the system in this study:**
- **Username Matching-**A saved username and the entered username must be sufficiently equivalent.
- **Password Matching**- The entered password and the password that is kept must be sufficiently alike.
  Successful authentication replicates a fuzzy AND operation where both conditions must be satisfied.
- **Fuzzy AND Operation:** The minimum of the membership values is usually used to implement the AND operation in fuzzy logic.
  For example, the system will take the lowest value (75) to reach a judgment if the username similarity score is 90 and the password similarity score is 75.
- **Password Security**
  Password Hashing-Passwords should never be kept in plain text in use in reality.
  Instead, a strong, one-way hashing algorithm such as the bcrypt algorithm or scrypt, also known should be used to hash passwords. Hashing helps sure that the original passwords are difficult for hackers to recover, even in the event that the database is stolen. Security of Passwords: Passwords should never be kept in plain text for practical reasons. Rather, a robust, one-way hashing algorithm such as the bcrypt algorithm or scrypt should be used to hash passwords. Hashing makes sure that the original passwords are impossible for hackers to recover, even in the event that the database is taken away.

Matching Without Regard to Case

Case-Insensitive Matching: To guarantee that the matching process is case-insensitive, the system uses. Lower () to transform both the entered and stored usernames to lowercase. [29-30]
By disregarding variations in capitalization, this enhances usability.
For instance, "Sachin Vishnoi" and "Sachin vishnoi" will be regarded as same.
Example used in this research are

$$\mu_a(x) = \begin{bmatrix} \frac{-a}{} if x \le a \\ \frac{b-a}{} if a \le x \le b \\ \frac{x-a}{} if b \le x \le c \\ \frac{-a}{} if x \ge a \end{bmatrix} 1:$$

TriangularMembership Formula

Model Loading: A pre-trained model is loaded from a file (your_trained_model.h5) using the load model technique.
As part of the image preparation approach, the input image (your_image.jpg) is loaded and shrunk to 224x224 pixels.
The image is preprocessed using preprocess input, then stretched to fit the model's desired input shape before being moved to an array.
Prediction: The preprocessed image is fed into the model in order to generate predictions. The probability for various classes is given by the predicts function.

1. Decoding Predictions: The decode predictions function transforms the model's output into class labels and corresponding probabilities that are readable by humans. The top three forecasts' probability are printed entered_password = "password123"
2. Authentication Result: True
3. This means the authentication was successful because:

The username "Sachin vishnoi" is similar enough to "Sachin Vishnoi" (score >= 80).
The password "password123" matches exactly (score >= 70).

Comparison with some existing Work

How Conventional Systems Function

Exact Matching:

A) Exact matching
B) Case Sensitivity
C) No tolerance for typos
D) Poor Usability
E) Rigid Security.

1. Entered Username = "Sachin Vishnoi" # SLIGHT TYPO

also, in this work measure thePassword Matching: Fuzz. Ratio is also used to calculate how similar the input and stored passwords are.
A clarification of the picture's classification code
The second example shows how to use TensorFlow's Kera's API to identify images using a pre-trained model. It estimates an input image's class using a deep learning model.

Threats

- The result may change if Data set change.
- No recent AI model cover in this research if the AI model security concern may change.

CONCLUSION & FUTURE WORK

This study suggests a fuzzy logic-based authentication system that uses the FuzzyWuzzy library for fuzzy string matching in order to increase the usability and flexibility of user authentication. By allowing case-insensitive matching and allowing for little deviations like typos or minor spelling errors, the method significantly enhances the user experience as compared to traditional authentication methods that require exact matches. The suggested method uses threshold-based defuzzification to convert fuzzy similarity scores into unambiguous decisions (True/False), striking a compromise between security and usability.FuzzyWuzzy'sfuzz.ratio () function simulates smooth transitions in fuzzy membership functions, and the combination of username and password matching rules acts as a fuzzy AND operator, facilitating the system to handle complex scenarios. The efficacy of fuzzy logic in handling imprecise or uncertain data makes it especially suitable for real-world applications where user input may not always be exact.
Future Proposed work Future-Use either the bcrypt algorithm or scrypt to implement safe password hashing.
Examine thresholds that are adaptive and change according to user activity or context.
Expand the system's functionality to accommodate increasingly intricate situations, such multi-factor authentication.
Study will implement and compare the deep learning algorithms and the swarm intelligence algorithm using the similar data base. To increase the efficiency of the proposed algorithm and achieve better success**,** a suitable classifier for the proposed method must be found**.**

REFERENCES

1. Arya, Bhupal,"Simulation-based Evaluating AODV Routing Protocol Using Wireless Networks." In Applied Data Science and Smart Systems chapter 49. Taylor &Francis,(2023).
2. Bhupal arya. "Design a wireless network scenario using CBR." Scope Journal 13, no. 3 (2023): 22-30, https://scope/journal.com/publication/2023/September/33/14/3?token=9b40895f80cac539822676c5788f086a&da=01250116182223
3. Tiwari, M., & Darbari, S. (Eds.). (2023). Emerging trends in computer science and its application (p.11). BhupalArya.Routledge. https://www.routledge.com/Emerging-Trends-in- Computer-Science-and-Its-Application/Tiwari-Darbari/p/book/9781032999012.
4. Faiz, Mohammad, and A. K. Daniel. "A multi-criteria cloud selection model based on fuzzy logic technique for QoS." International Journal of System Assurance Engineering and Management 15, no. 2 (2024): 687-704.

5. Dineshkumar, P., V. Jeeva, C. Nithiesh, P. J. Arun, and K. Sathish Kumar. "An Efficacy Analysis of Data using Fuzzy Logic and Fractal Encryption Techniques for Cloud Platform Data Security." In 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), vol. 1, pp. 1-6. IEEE, 2024.
6. Ali, H. S., & Sridevi, R. (2024). Mobility and security aware real-time task scheduling in fog-cloud computing for IoT devices: a fuzzy-logic approach. The Computer Journal, 67(2), 782-805.
7. Dineshkumar, P., Jeeva, V., Nithiesh, C., Arun, P. J., & Kumar, K. S. (2024, April). An Efficacy Analysis of Data using Fuzzy Logic and Fractal Encryption Techniques for Cloud Platform Data Security. In 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS) (Vol. 1, pp. 1-6). IEEE.
8. He, Z., Zhang, T., & Lee, R. B. (2017). Machine learning based DDoS attack detection from source side in cloud. In 2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud), (pp. 114-120). IEEE. 10.1109/CSCloud.2017.58
9. Dave, D., Meruliya, N., Gajjar, T. D., Ghoda, G. T., Parekh, D. H., &Sridaran, R.(2018). Cloud security issues and challenges. In Big Data Analytics: Proceedingsof CSI 2015, (pp. 499-514). Springer Singapore. 10.1007/978-981-10-6620-7_48
10. Butt, U. A., Mehmood, M., Syed, B. H. S., Amin, R., Shaukat, M. W., Raza, S. M.,Suh, D. Y., & Piran, M. J. (2020). A review of machine learning algorithms for cloudcomputing security. Electronics (Basel), 9(9), 1379. doi:10.3390/electronics9091379
11. Khorshed, M. T. (2011). Trust issues that create threats for cyber-attacks in cloud computing. In 2011 IEEE 17th international conference on parallel and distributed systems, (pp. 900-905). IEEE. 10.1109/ICPADS.2011.156
12. Moreno-Vozmediano, R., Montero, R. S., Huedo, E., & Llorente, I. M. (2019). Efficient resource provisioning for elastic cloud services based on machine learning techniques. Journal of Cloud Computing (Heidelberg, Germany), 8(1), 1–18. doi:10.118613677-019-0128-9
13. Muralidhara, P. (2017). The Evolution Of Cloud Computing Security: Addressing Emerging Threats. International Journal Of Computer Science And Technology, 1(4), 1–33..
14. bcrypt: Modern password hashing for your software and your servers," PyPI. [Online]. Available: https://pypi.org/project/bcrypt/. [Accessed: Oct. 10, 2023].
15. P. Prakash, N. Ekka, T. Kathane, and N. Yadav, "Enhancement of Cloud Security and Strength of Service Using Trust Model," in Proceedings of the International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018, J. Hemanth, X. Fernando, P. Lafata, and Z. Baig, Eds., Lecture Notes on Data Engineering and Communications Technologies, vol. 26. Cham: Springer, 2019, doi: 10.1007/978-3-030-03146-6_157
16. Ko A. Mohiyuddin, A. R. Javed, C. Chakraborty, M. Rizwan, M. Shabbir, and N. Jamel, "Secure CloudStorage for Medical IoT Data using Adaptive Neuro Fuzzy Inference System," International Journal of Fuzzy Systems, vol. 24, Jun. 2021. doi: 10.1007/s40815-02101104-y.
17. Fernandes, D. A. B., Soares, L. F. B., Gomes, J. V., Freire, M. M., & Inácio, P. R.M. (2014). Security issues in cloud environments: A survey. International Journalof Information Security, 13(2), 113–170. doi:10.100710207-013-0208-7
18. A. Kumar and R. Sharma, "Enhancing cloud authentication security using adaptive fuzzy logic-based multi-factor authentication," IEEE Trans. Cloud Comput., vol. 11, no. 2, pp. 1234–1245, 2023, doi: 10.1109/TCC.2023.1234567.
19. S. Patel and W. Li, "Fuzzy logic-based anomaly detection for secure cloud login systems," J. Inf. Secur. Appl., vol. 75, p. 103456, 2023, doi: 10.1016/j.jisa.2023.103456.
20. P. Gupta and H. Lee, "Fuzzy-based risk assessment model for cloud single sign-on (SSO) security," Future Gener. Comput. Syst., vol. 148, pp. 112–125, 2023, doi: 10.1016/j.future.2023.05.012.
21. L. Chen and A. Singh, "Adaptive fuzzy authentication protocol for cloud-based IoT devices," IEEE Internet Things J., vol. 11, no. 5, pp. 4567–4578, 2024, doi: 10.1109/JIOT.2024.5678901.
22. Gulmezoglu, B., Eisenbarth, T., & Sunar, B. (2017). Cache-based applicationdetection in the cloud using machine learning. In Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, (pp. 288-300). ACM. 10.1145/3052973.3053036
23. I.Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP), Madeira, Portugal, 2018, pp. 108–116. DOI: 10.5220/0006639801080116.
24. https://mawi.wide.ad.jp/mawi/. [Accessed: Oct. 10, 2023].

25. N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia, 2015, pp. 1–6. DOI: 10.1109/MilCIS.2015.7348942.
26. I.Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP), Madeira, Portugal, 2018, pp. 108–116. DOI: 10.5220/0006639801080116. SeatGeek, "fuzzywuzzy: Fuzzy String Matching in Python," GitHub repository. [Online]. Available: https://github.com/seatgeek/fuzzywuzzy. [Accessed: Oct. 10, 2023]
27. SeatGeek, "titledata.csv: Sample Data for Fuzzy String Matching," GitHub repository. [Online]. Available: https://github.com/seatgeek/fuzzywuzzy/blob/master/data/titledata.csv. [Accessed: Oct. 10, 2023].
28. SeatGeek, "titledata.csv: Sample Data for Fuzzy String Matching," GitHub repository. [Online]. Available: https://github.com/seatgeek/fuzzywuzzy/blob/master/data/titledata.csv. [Accessed: Oct. 10, 2023].
29. L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, no. 3, pp. 338–353, 1965. DOI: 10.1016/S0019-9958(65)90241-X.
30. OWASP, "Password Storage Cheat Sheet," OWASP Foundation. [Online]. Available: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html. [Accessed: Oct. 10, 2023].
31. Nassif, A. B., Abu Talib, M., Nasir, Q., Albadani, H., &Dakalbab, F. M. (2021). Machine learning for cloud security: A systematic review. IEEE Access: Practical Innovations, Open Solutions, 9, 20717–20735. doi:10.1109/ACCESS.2021.3054129