Round Trip Latency Based Authentication Scheme in Fog-Enabled Cloud Computing System

C. Nagarani, R. Kousalya

Abstract: ***In fog-enabled cloud computing framework, one of the most challenges is security promises due to the compromised passwords. To overcome this issue, different multifactor authentication schemes have been developed that requires additional authentication credentials along with the standard password to authenticate the login. Among those schemes, Communication Latency-based Authentication Scheme (CLAS) increases the protection of conventional web authentication schemes by leveraging the Round Trip network communication Latency (RTL) between clients and authenticators together with standard password. It uses RTL of clients to secure against password compromise. On the other hand, it can support compromise of either the password or the profiled location of a user. This makes it susceptible to same location attacks. As a result, an integration of additional profiling features is needed to attain more robust and flexible defense against password compromise. Hence in this paper, an extended CLAS is proposed that mainly investigates the mobility and same location challenges in CLAS. Initially, the legitimate login failures are solved by handling both selective and arbitrary mobility of users. For selective mobility case, CLAS generates an individual profile for each location and the user may be granted access if his/her real-time login profile matches any of the stored reference profiles. For arbitrary mobility case, CLAS is integrated with two-factor authentication mechanism to authenticate the user. In addition, the defense against Mimic attacks is improved by utility metric-based location anonymization and obfuscation of RTL algorithms. By using these algorithms, the user's locations are anonymized and the values of RTL are obfuscated to defend against user compromise attempts for impersonating the RTL by getting nearer to the user location. Moreover, a keystroke dynamics measure is introduced with obfuscated RTL measures to effectively defend the same location attacks. This technique alleviates the potential impacts of network instabilities on RTL measurements. As well, it increases the authentication sample space and so improves the security guarantee of CLAS. Finally, the simulation outcomes illustrate that an extended CLAS has the ability to reduce both false positive and false negative rates.***

Index Terms: ***Fog computing, Cloud computing, CLAS, Dummy-based location anonymization, Two-factor authentication, Keystroke dynamics***

I. INTRODUCTION

Fog computing is a decentralized computing framework where the information is processed and stored between the source and a cloud architecture. This framework can reduce the need of processing and storing large number of redundant data.

Due to this, the data transmission overhead is reduced and the performance of cloud computing is improved efficiently. Mostly, the fog computing paradigm is encouraged by the rapid growth of Internet-of-Things (IoT) devices. When using a normal client-server framework, there may be challenges in scalability and reliability due to high overload in the server. These challenges can solve by the fog paradigm which provides the scalable decentralized solution. This can be achieved by the novel hierarchically distributed and local platform known as fog computing between the cloud system and end-user devices [1-3]. Generally, a fog system has its own benefits such as it requires relatively less computing resources for memory and storage, high ability to process the data from different set of devices, etc. Owing to fewer requirements of resources, it may have a complexity to carry out a complete set of defence solutions for attacks detection and mitigation. On the other hand, there are no precise defence certifications and measures for this framework. Also, authentication and authorization solutions are not appropriate for this platform since fog devices are operating at the edge of networks. The functioning settings of fog devices may experience with several intimidations that do not present in a well-managed cloud.

Normally, fog devices have various sort of connectivity to the secluded cloud authentication server which is used for distributing the authentication data and collecting audit logs. Nevertheless, this connectivity may be measured in certain settings such as smart grid. Probably, an authentication protocol such as isolated authentication dial in user service or lightweight directory access control over this link is not well-known [4]. Further, trust on cloud central authentication servers is risky since authentication must prolong to apply for personnel accessing devices locally when isolated authentication server communications are lost. A condition to guarantee that basic access presented in crisis circumstances may be significant, even if it means bypassing usual access control but with an audit trail.

The most thriving attacks utilize authentication recommendations. Passwords have been the most disreputable, but foremost authentication recommendation for web services. Nonetheless, attackers constantly innovate ways to compromise the passwords. Such password evolving weakness is tackled by multifactor authentication schemes [5]. Generally, it requires other authentication codes with usual password to login. Though these schemes significantly improve the protection of password-based

Revised Manuscript Received on July 06, 2019.

C. Nagarani, Department of Computer Science, PSG College of Arts and Science, Coimbatore, Tamilnadu, India.

R. Kousalya, Department of Computer Application, Dr. N. G. P. Arts and Science College, Coimbatore, Tamilnadu, India.

systems, they tolerate from several restrictions and new susceptibilities. The attacker may easily present the user with a counterfeit deceptive screen to input his/her extra codes which the invader utilizes in synchronized for imitating the genuine users. As a result, CLAS has been proposed [6] that enhances the security of conventional multifactor authentication schemes by levering the RTL between clients and authenticators. CLAS profiles RTL along with the conventional credentials of clients and utilizes them to secure against password compromise. Also, it restricts the login to profiled locations whilst challenging extra data for non-profiled ones which greatly reduces the attack plane even when the genuine recommendations are compromised. However, it requires an integration of additional profiling features to attain more strong and flexible defense against password compromise.

Hence in this article, an extended CLAS is proposed that mostly aims the mobility and similar location challenges in the conventional CLAS. Initially, a two-factor authentication mechanism is integrated with the CLAS to manage both mobility and valid login failures. In this technique, two types of mobility patterns are considered such as selective and arbitrary. For selective mobility case, CLAS simply generates an individual profile for each location to grant access by comparing his/her real-time login profile with any of the stored reference profiles. For arbitrary mobility case, CLAS is integrated with two-factor authentication mechanism. As well, utility metric-based location anonymization and obfuscation of RTL algorithms are proposed to improve the defense against Mimic attacks. Based on these algorithms, the user's locations are anonymized and the RTL values are obfuscated that preserves the user compromise attempts from impersonating the RTL by getting nearer to the user location. Further, a keystroke dynamics measure is introduced with obfuscated RTL measures to effectively defend the same location attacks. Therefore, this incorporation can enhance the authentication sample space such that the security guarantee of CLAS is further improved.

The remaining of the article is structured as follows: Section II presents the literature survey interrelated to the authentication and authorization schemes in fog-enabled cloud computing. Section III explains the proposed methodology. Section IV illustrates the simulation results of the proposed scheme. Finally, Section V concludes the research work.

II. LITERATURE SURVEY

A novel methodology [7] was proposed to realize the Ciphertext-Policy Attribute-Based Encryption (CP-ABE) technique by using three constructions. The first system was used to allow an encryption algorithm that specifies an access formula. An access control was expressed by a Linear Secret Sharing Scheme (LSSS) matrix over the attributes in the system. Also, the other two constructions were providing the tradeoff of some performance parameters for provable security under the respective decisional Parallel Bilinear Diffie-Hellman Exponent (d-PBDHE) assumptions. However, the assumptions were still not-interactive.

A secure and efficient mutual authentication mechanism [8] was proposed for the edge-fog-cloud network in order to mutually authenticate the fog users at the edge of the network with the fog servers at the fog layer. In this mechanism, the fog users have the ability to mutually authenticate with new fog servers connecting with the network, without the requirement of re-registering and additional overheads. Also, the servers on the fog were needed to store only one secret key for each fog user. Conversely, the fog users were entirely isolated to any public-key infrastructure. However, this mechanism was not suitable for large-scale networks.

A model of Anonymous and Secure Aggregation Scheme (ASAS) [9] was proposed in fog-based public cloud computing. In this model, the data from the terminal node was aggregated and transmitted to the Public Cloud Server (PCS) by the fog node. By using this scheme, the fog node can support terminal devices to upload their data to PCS. Also, it can preserve bandwidth between the fog node and PCS. Meanwhile, it not only protects the identities of terminal devices by using pseudonyms; but it ensures data secrecy via a homomorphic encryption technique. However, the communication overhead was high.

A CP-ABE technique [10] was proposed against key-delegation abuse in fog computing. In this technique, the property of bilinear groups was used. Each client private key has elements for all features was generated based on either set of group elements according to if the user owns this attribute or not. After, the secret sharing scheme was applied on all attributes and the bilinear map of key components and corresponding ciphertext components for all attributes were forced. As a result, the key cannot be split nor combined with the other private keys. However, the computation cost during encryption process was high. An efficient key exchange protocol [11] was proposed based on the CP-ABE technique to establish the secure communications among the users. This protocol was mainly proposed for encrypted key exchange based on CP-ABE that combines encryption and signature for achieving a fine-grained data access control, confidentiality, authentication and verifiability. Also, the security of this protocol was investigated under different attack scenarios. However, the computation overhead was high and also an efficiency of access structure was less.

An analysis of location authentication problem in fog computing [12] was presented by creating a secure positioning protocol with neighbourhood confidentiality in the encircled recovery model. Initially, an investigation was carried out to explore how to describe neighbourhood confidentiality that requires almost all parties except the prover along verifiers whereas the outside invaders do not have the ability to discover any extra data about an accurate location of the prover for secure positioning protocol in the encircled recovery model. After, a secure positioning protocol was constructed in the one-dimension and 3-dimension settings. Moreover, advanced cryptographic protocols were constructed to exploit the location verification. However, this protocol can only promise the neighbourhood secrecy if all invaders are not at a half-line from the claimed region via a verifier.

III. PROPOSED METHODOLOGY

In this section, the proposed scheme is described in brief. Typically, the CLAS [6] consists of client, server and Stealthy Relay (SR). The CLAS is understandable to end clients since they do not grant any other authentication data beyond that of username and password. User's login to servers for accessing services and a server commences the process to create his/her profile. For each client in CLAS, the profile is the mean and standard deviation of the RTL between the client and the server via a SR. The observed RTL between client and server approximately pursues a Gaussian distribution to analyze the minimum requirements for constructing the agent profiles whilst handling bandwidth and login latency overhead. Servers have SR in the round trip path to users for thwarting potential efforts to discover profile parameters. SR achieves the learning of RTL by generating new path-segments in the round trip route between the server and its clients. The value of RTL varies when any of the clients modifies the locality of its Internet access. Once RTL measure is completed, profile parameters are stored on the server together with the fixed recommendations for upcoming login authentication. After that, login efforts are profiled in real-time and compared against the stored profile parameters. When the partial real-time profile parameters appear in the fixed margins of the stored profile parameters, the access is granted. The RTL is securely measured as follows:

$$RTL = T_{rcv} - T_{send} \qquad (1)$$

In the equation (1), T_{rcv} is the time taken to receive the user acknowledgement at the server and T_{send} is the transmit time of each downstream profiling signal. Conversely, the RTL is defined as the total delay over the path-segment from the server to the SR (D_{SS}), the delay over the path-segment from the SR to the client (D_{SU}) and the delay over the path-segment from the client to the server (D_{US}). The attackers have the ability to estimate D_{US} by ping the server from the locality of the client whereas it is not feasible for him/her to estimate D_{SU} and D_{SS} due to the one-way communication framework of the SR. To annihilate this ability of attackers, the locations of the user need to be anonymized.

As a result, the proposed extended CLAS is proposed in which the user's locations are anonymized to defend the mimic attack. Normally, attackers try to get similar RTL by getting nearer to the location of the user. So, utility metric-based location anonymization is proposed to anonymize the user's location and array index transformation using composite functions is proposed to obfuscate the RTL values. Based on these, the mimic attacks can be easily defended even if attackers find the user's location and get near to it. Therefore, a high level of security can be guaranteed. Also, the legitimate user login failures are effectively solved by handling the mobility and similar locality challenges in CLAS. Moreover, the potential integration of extra profiling features such as keystroke dynamics is proposed to attain more strong and flexible mitigation against password compromise. Fig. 1 shows the registration phase and authentication phase in the extended CLAS.

A. Handling Mobility and Legitimate Login Failures

Normally, CLAS recognizes the clients based on the mean and the standard deviation of the RTL which is extremely dependent on the login location of the client. If two clients are connected to the similar local area network part or connected to the similar access point or linked to the similar network cell, then both users have the same location. Therefore, if a genuine client's login from a locality other than the profiled one, he/she may be denied access with a higher probability. Also, the client may not succeed to login from his/her profiled locality due to network instabilities. To tackle this condition, the solutions with CLAS are proposed that effectively handles both mobility and valid login malfunctions.

- ***Selective Mobility*:** It is the case in which a client repeatedly login from a set of locations like home, organization and library. In this situation, a separate profile for each locality is created by CLAS. For each new location, the service provider is registered with the user and a new reference profile is also built by the server. After, the profile parameters for each new location are stored on the server together with the conventional recommendations for upcoming login authentication. Moreover, a client login will be approved if his/her real-time login profile matches any of the stored profiles. Thus, the legitimate login failures are neglected. Due to this process, the defence advantages of CLAS i.e., resiliency to phishing, etc., are enhanced.
- ***Arbitrary Mobility*:** It is the common mobility model in which the clients may login from random locations. To handle this situation, CLAS is integrated with two-factor authentication mechanism. Generally, it provides a two-way authentication to the user. Authentication depends on the username, password and a security token. The user utilizes one of his/her profiled locations for obtaining the temporary token that can used to login from a new profiled locality. Initially, user's login from one of his/her profiled locations and requirements a temporary token to the client. Then, the client may give the temporary token as a second distribution code used by the server to bypass the regular profile authentication. This permits the client to login from any locality during the authentication of the temporary token. Therefore, a legitimate user login failure can be reduced by two-factor authentication.

B. Improved Defense against Mimic Attacks

The anonymous region is the size of the minimum convex covering all locations included in a service request from user to the server. The user location anonymization mechanism, namely utility metric-based location anonymization method is proposed that consists of three phases such as the region partition phase, the region growth phase and the region reduction phase. Initially, it creates a full-anonymized area where all clients stay alive and recursively partitions the anonymized region until the resulted anonymized regions do not satisfy (k, w)-anonymity i.e., k or more clients presented in an anonymized locality with probability superior than or equivalent to w. Consider U is the set of users and $L(u)$ is the anonymized region where

user u existed at the certain time. Take $p_{u,L}$ be the probability that user u exist in region L at the particular period and g_u is the survival probability circle of client u. The value of $p_{u,L}$ is computed as follows:

$$p_{u,L} = \frac{|L \cap g_u|}{|g_u|} \quad (2)$$

In equation (2), $L \cap g_u$ is the overlapped region of L and g_u. Then, the utility is defined by using the following equation:

$$Utility = \sum_{u \in U} \frac{(p_{u,L(u)})^{\alpha}}{|L(u)|} \quad (3)$$

In equation (3), α is used for adjusting the weight of the size of the anonymized regions and the chance of clients active in the related anonymized region.

- ***Region Separation Phase:*** The complete-anonymized region is separated iteratively. Each iteration selects the longer dimension to be separated and separates the selected dimension. Therefore, each separated region has half of the users. Consider U_L is the set of clients who present in region L with probability larger than 0. The probability $P(L,k)$ that k or more users exist in region L is computed as follows:

$$P(L,k) = 1 - \sum_{\{L|L \in T(U_L) \wedge |L| < k\}} \left[\prod_{u \in L} p_{u,L} \cdot \right.$$
$$\left. u \notin L \wedge u \in UL1 - pu,L \right. \quad (4)$$

In equation (4), $T(U_L)$ is the power set of U_L and $|L|$ is the size of region L. While the size of U_L is high, the number of computation becomes very heavy. To reduce the number of computation significantly, the semi-optimal algorithm is used. At first, identify that (k,w)-anonymity is not satisfied when $|U_L| < k$. After that, calculate the number of probability survival circles included fully in anonymized region L and consider m is the number. In this case, the region L satisfies (k,w)-anonymization while the probability that $k-m$ or more users excluding for m users and existing region L is superior than or equivalent to w.

Further, the value of probability that each user normally exists in region L is discretized into d levels. Consider c_j is the number of clients for whom $p_{u,L}$ is superior than or equivalent to j/d and less than $(j+1)/d$. The value of c_j is computed as,

$$c_j = \left| \left\{ u | u \in U \wedge \frac{j}{d} \leq p_{u,L} < \frac{j+1}{d} \right\} \right| \quad (5)$$

The probability $P(L,k)$ that k or more clients present in region L has the following condition:

$$P(L,k) \geq 1 - \sum_{q=0}^{k-m-1} Q(q) \quad (6)$$

Where

$$Q(q) = \sum_{i_1=\underline{\Lambda}(1)}^{\overline{\Lambda}(1)} \cdot \sum_{i_2=\underline{\Lambda}(2)}^{\overline{\Lambda}(2)} \cdots \sum_{i_{d-1}=\underline{\Lambda}(d-1)}^{\overline{\Lambda}(d-1)} \cdot \prod_{j=1}^{d-1} \left[c_j C_{i_j} \cdot \right.$$
$$\left. jdij \cdot 1 - jdcj - ij \right.$$

$$\overline{\Lambda}(s) = min\left(c_s, q - \sum_{j=1}^{s-1} i_j\right)$$
$$\underline{\Lambda}(s) = max\left(0, q - \sum_{j=1}^{s-1} i_j - \sum_{j=s+1}^{d-1} c_j\right)$$

Consider L_1 and L_2 are the two regions generated by separating one region. When the condition $P(L_1,k) < w$ or $P(L_2,k) < w$ holds, the separation is cancelled.

- ***Region Expansion Phase:*** In the region separation phase, two anonymized regions L_1 and L_2 are obtained. Then, those two regions are expanded separately. The aim of this phase is expanding the margin area to the location that can attain the maximum utility. As the anonymized region L_1 only expands, it will satisfy (k,w)-anonymity before completing this phase. Consider the region that contains L_1 and the existence probability circles of all the clients are included in region L_1. Also, consider $L_{1,max}$ is the expanded region and A_1 is the location of the expanded boundary side. After that, the optimal location of the boundary side is searched that maximizes the utility.

The probability that every client normally presents in region L_1 increases by expanding the boundary side. But, the increasing rate can be reduced by gradually expanding the margin area. Hence, the utility function is a unimodal function that represents the location of the margin area in the range from A_0 and A_1. The maximum value of the unimodal function is obtained by using the golden section search [13]. After that, the locations A_{n1} and A_{n2} are defined as follows:

$$A_{n1} = \frac{\phi \cdot A_0 + A_1}{\phi + 1}, \; A_{n2} = \frac{\phi \cdot A_1 + A_0}{\phi + 1} \quad (7)$$

In equation (7), ϕ denotes the golden ratio and $\phi = 1.618$. Consider the utility with location A_{n1} is smaller than that with location A_{n2}. In this case, the margin area of L_1 is updated from A_0 to A_{n1}. If the utility with location A_{n2} is smaller than that with location A_{n1}, the boundary side of L_{max} is updated from A_1 to A_{n2}. This process is continued based on the updated L_1 and L_{max}. At last, the optimal location of the margin area is obtained that increases the utility. Also, this process is conducted for L_2 and the optimal location of the boundary side of L_2 is found out between A_0 to A_2 in the similar manner.

- ***Region Reduction Phase:*** The above two phases are continued until the resulting anonymized regions do not assure (k,w)-anonymity and then the region reduction stage is performed. For each anonymized region, the process of the region reduction stage is executed. Consider L is one of the anonymized regions L_{min} is the minimum region that intersects the existence probability circles of all clients in L. The goal of this phase is obtaining the optimized region that increases the utility within the range of L to L_{min}. The anonymized region L is fine-tuned by moving one of its four sides. Then, two locations are computed for fair segment search for each region. As well, $P(L,k)$ and the utility for each location of each side are computed. After that, the location is adopted that satisfies (k,w)-anonymity and maximizes the utility. This process is continued until the optimal anonymized region is obtained that maximizes the utility. Based on this location anonymization method, the information associated with the user's locations are anonymized so that the anonymized region contains k or more users.

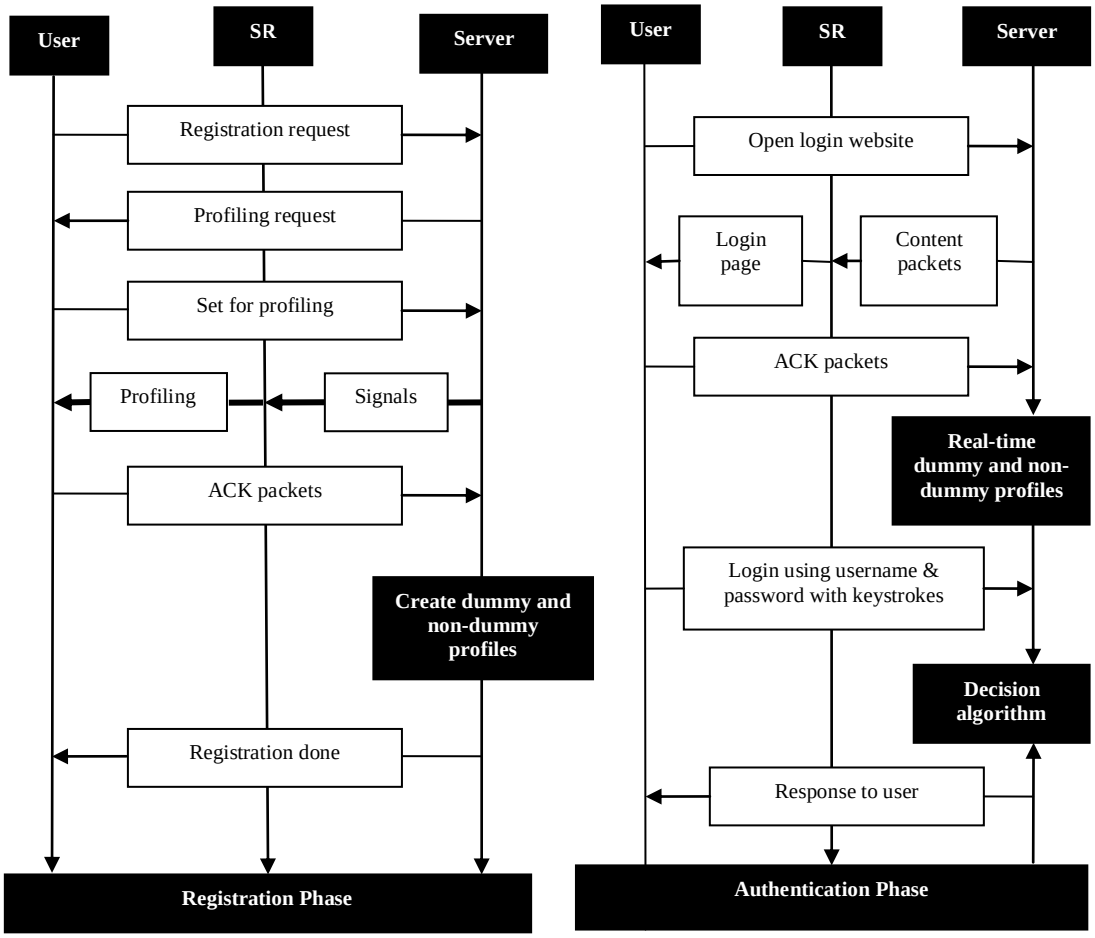

Fig.1: Extended CLAS Authentication Flowchart

Algorithm for Anonymization for (k, w)-Anonymity

Input: Parameters k and w, Target Region L, Set of users U

Output: Set of anonymized regions $\mathfrak{L}$

$//separation\&expansion(k, w, L, U)$

$\mathfrak{L} \Leftarrow \emptyset;$

$d \Leftarrow selectDimension(L);$

$A_0 \Leftarrow obtainMedianPoint(L, d, U);$

$L_1 \Leftarrow$ *one of the two regions generated by separating L at* A_0;

$L_2 \Leftarrow another\ region;$

$if(both\ L_1\ \&\&\ L_2\ satisfy\ (k, w) - anonymity)$

{

$L_1 \Leftarrow regionExpansion(L_1, U);$

$L_2 \Leftarrow regionExpansion(L_2, U);$

$\mathfrak{L} \Leftarrow \mathfrak{L} \cup separation\&expansion(k, w, L_1, U);$

$\mathfrak{L} \Leftarrow \mathfrak{L} \cup separation\&expansion(k, w, L_2, U);$

}

$else$

{

$\mathfrak{L} \Leftarrow \{L\};$

}

$for(L \in \mathfrak{L})$

{

$L' \Leftarrow regionReduction(k, w, L, U);$

$\mathfrak{L} \Leftarrow (\mathfrak{L} \backslash \{L\}) \cup \{L'\};$

}

$return\ \mathfrak{L};$

- ***Array Index Transformation based RTL Obfuscation:*** In addition to the user's location anonymization, the measured RTL values are obfuscated to make attackers not possible to estimate the values of D_{US}. Normally, the obfuscation is used to protect the fog network from attackers. The, obfuscation of RTL is done based on the array index transformation using composite functions. Once the RTL values (X) are measured, they are stored in an array. Consider $X = f(x) = 2 * x + 3$ is the function representing the new value of X. Consider $Y = g(X) = f\left({}^{(X-3)}/_{2}\right)$ is a function representing the new position of i^{th} element in the stored array. Therefore, member variable x can be shown as a composite function of $f(g(x))$. The results tabulated in Table 1 show that the value of x remains similar before and after obfuscation.

Adjustment to the value of the member variable X will require to be made based on the functions f and g. For instance, the value X in Table 1 is incremented by 4 in the obfuscated program unlike in the original one the region separation phase, two anonymized regions L_1 and L_2 are obtained. Then, those two regions are expanded.

Table 1: Variations in $x, X = f(x), Y = g(X)$

Before	After
$int\ x = 1;$	$int\ X = f(x);$
$while(x < 1000)$	
$while(X < f(1000))$	

{ $A[x]; x++;$ } { $A[g(X)]; X = X + 2;$ }

x	$X = f(x) = 2 * x + 3$	$Y = g(X) = f\left((X-3)/2\right)$
2	7	2
4	11	4
6	15	6
8	19	8

C. Improved Defense against Same Location Attacks

The proposed extended CLAS can endure compromise of both the password and the profiled location of a user to mitigate the same location attacks by integrating an additional profiling feature such as keystroke dynamics with the RTL. When a user attempts login, it passes the RTL and keystroke dynamics check, extended CLAS may raise an alert and request the user to answer a few security questions or authenticates by using a second channel. Since keystroke dynamics which is exhibited in a user's typing patterns offers an incomparable signature for authenticating the client. Latency between successive keystrokes, keystroke lengths, finger position and applied force on the keys may be utilized for constructing a unique signature per individual.

Keystroke dynamics is the task of analyzing the user types at a terminal by observing the keyboard inputs thousands of times per seconds in an effort for identifying user's based on normal typing rhythm patterns. It uses two orthogonal components such as total time in which the first key is dejected i.e., keystroke length and the time between a key is released and the subsequent key is pressed i.e., keystroke latency. For each registration samples, consider a user name/password is represented by the following sequence:

$$P_1, R_1, P_2, R_2, \dots, P_n, R_n \quad (8)$$

Here, P_n and R_n are the press and release time of n^{th} keystroke of a user name/password. The elements of feature vectors extracted from the original keystroke information are spilt into two types such as dwelling time and flight time. The dwelling time is computed by $R_n - P_n$ and the flight time is calculated by $P_n - R_{n-1}$. Thus, the extracted feature from the original sequence is represented as follows:

$$G = (R_1 - P_1, P_2 - R_1, R_2 - P_2, \dots, P_n - R_{n-1}, R_n - P_n) \quad (9)$$

Then, for each keystroke dynamics, their mean and deviation is calculated whereas the outliers are neglected for efficient authentication. As a result, the server and the user are intended to measure their obfuscated RTL and keystroke dynamics including mean and standard deviations of all the profiling contain the client profile. During the authentication stage, the login packets from the server are recognized by the user. The mean and standard deviation of the RTL and keystroke dynamics in real-time is computed by the server by using the profiling signals and acknowledgements. The observed both RTL and keystroke dynamics between user/client and server approximately follows a Gaussian distribution. After that, the server evaluates the real-time measured profile parameters with those of the reference profile based on the predetermined threshold value. According to the obtained result of the evaluation, the access is either approved or denied.

IV. RESULTS AND DISCUSSION

In this section, the performance of extended CLAS is evaluated by implementing fog-enabled cloud architecture using JAVA in which the user credentials are stored in the Microsoft Access database. The SR functionality is performed by configuring and deploying the PC [6]. In this experiment, 175 users are considered to create and profile a user account. After that, each user is set to play attacker's role who tries to login using the credentials of the remaining 174 users. Each attacking user launches 310 attacks against each of the remaining users. Each unsuccessful login is retried for a maximum of 3 times. The evaluation is mainly based on the below-mentioned metrics:

- False Negative Rate (FNR): It is the possibility that a legitimate user fails to login from his/her profiled location.
- False Positive Rate (FPR): It is the probability that an executor who possesses legitimate user credentials successfully authenticates on behalf of the legitimate user.
- Login latency overhead: It is an additional time taken to successfully authenticate a user by using extended CLAS.
- Bandwidth and storage overhead: It is an additional network bandwidth and the storage required by extended CLAS.

Mainly, the extended CLAS has three parameters such as the Error Tolerance (ET), number of attackers and the highest amount of unsuccessful Login Retries (LR). Both ET and LR impact the trade-off between FNR and FPR.

Table 2 shows the comparison of proposed and existing techniques in terms of FNR and FPR.

Table 2: Comparison of FNR and FPR

No. of Attackers	CLAS		Extended CLAS	
	FNR	FPR	FNR	FPR
50	0.007	0.0017	0.005	0.0011
100	0.01	0.002	0.008	0.0015
150	0.013	0.0023	0.011	0.0019
200	0.016	0.0026	0.014	0.0023
250	0.019	0.0029	0.017	0.0027
300	0.022	0.0032	0.02	0.003

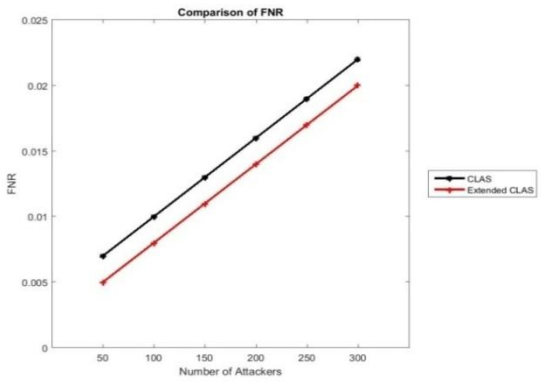

Fig.2: Comparison of FNR

Fig. 2 shows the comparison of FNR for extended CLAS and CLAS.

In this graph, x-axis denotes the number of attackers and y-axis denotes the FNR. Through this analysis, it is identified that the proposed extended CLAS achieves the lower FNR which refers the probability that a legitimate client not succeed to login from his/her profiled location is significantly reduced.

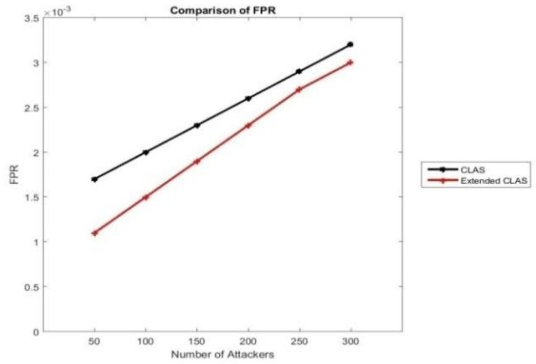

Fig.3: Comparison of FPR

The comparison of FPR for extended CLAS and CLAS is shown in Fig. 3. In this graph, x-axis denotes the number of attackers and y-axis denotes the FPR. From this analysis, it is recognized that the proposed extended CLAS achieves the lower FPR which refers the probability that an executor who acquires valid client recommendations effectively validates on behalf of the genuine client are reduced.

Table 3 shows the comparison of proposed and existing techniques in terms of login latency overhead and bandwidth overhead.

Table 3: Comparison of Login Latency Overhead and Bandwidth Overhead

No. of Attackers	Login Latency Overhead		Bandwidth Overhead	
	CLAS	Extended CLAS	CLAS	Extended CLAS
50	0.2	0.14	13	9
100	0.26	0.2	17	12
150	0.32	0.26	21	15
200	0.38	0.32	25	18
250	0.44	0.38	29	21
300	0.5	0.44	33	24

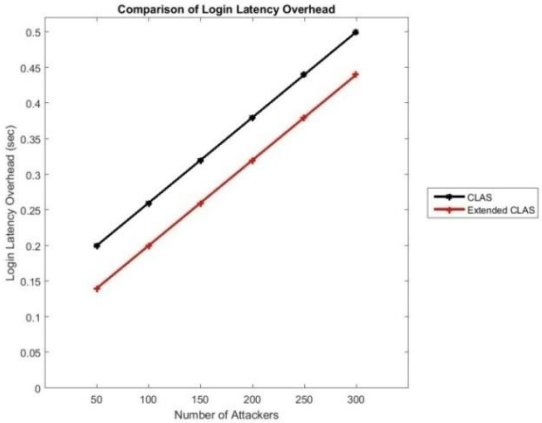

Fig.4: Comparison of Latency Overhead

Fig. 4 illustrates the comparison of login latency overhead for the proposed and existing schemes. In this graph, x-axis denotes the number of attackers and y-axis denotes the login latency overhead taken in seconds. From this analysis, it is observed that the proposed extended CLAS achieves the lower login latency overhead which refers the extra time required to authenticate the user is decreased significantly.

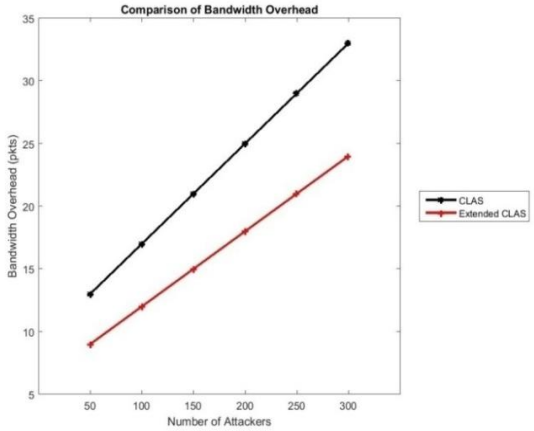

Fig.5: Comparison of Bandwidth Overhead

Fig. 5 illustrates the comparison of bandwidth overhead for the proposed and existing schemes. In this graph, x-axis denotes the number of attackers and y-axis denotes the bandwidth overhead taken in packets. From this analysis, it is observed that the proposed extended CLAS achieves the lower bandwidth overhead which refers the extra number of login packets required to authenticate the user is decreased. It states that the required number of profiling signals and login packets are sufficient to built user's profiles. Likewise, the storage overhead is also reduced since the credential database stores the profile parameters using floating point representation.

Table 4-6 shows the comparison of proposed and existing techniques in terms of ROC when varying number of attackers, ET and LR, respectively.

Table 4: Comparison of ROC when varying Number of Attackers

FNR	FPR					
	CLAS			Extended CLAS		
	N=30	N=70	N=110	N=30	N=70	N=110
50	0.011	0.006	0.003	0.008	0.004	0.001
100	0.01	0.005	0.001	0.007	0.003	0
150	0.009	0.003	0	0.006	0.002	0
200	0.007	0.002	0	0.005	0.001	0
250	0.006	0.001	0	0.004	0	0
300	0.005	0	0	0.003	0	0

Table 5: Comparison of ROC when varying ET

FNR	FPR					
	CLAS			Extended CLAS		
	ET=0.2*S	ET=0.3*S	ET=0.4*S	ET=0.2*S	ET=0.3*S	ET=0.4*S
50	0.0018	0.006	0.011	0.0013	0.004	0.009
100	0.0021	0.008	0.013	0.0017	0.006	0.011
150	0.0026	0.010	0.015	0.0022	0.008	0.013
200	0.0036	0.012	0.017	0.0029	0.010	0.015
250	0.0044	0.014	0.019	0.0036	0.012	0.017
300	0.0056	0.016	0.021	0.0041	0.014	0.019

Table 6: Comparison of ROC when varying LR

FNR	FPR					
	CLAS			Extended CLAS		
	LR=1	LR=2	LR=3	LR=1	LR=2	LR=3
50	0.026	0.023	0.02	0.024	0.022	0.019
100	0.021	0.019	0.017	0.02	0.018	0.016
150	0.017	0.015	0.013	0.016	0.014	0.012
200	0.014	0.011	0.009	0.012	0.01	0.008
250	0.009	0.007	0.005	0.008	0.006	0.004
300	0.006	0.004	0.002	0.005	0.003	0.001

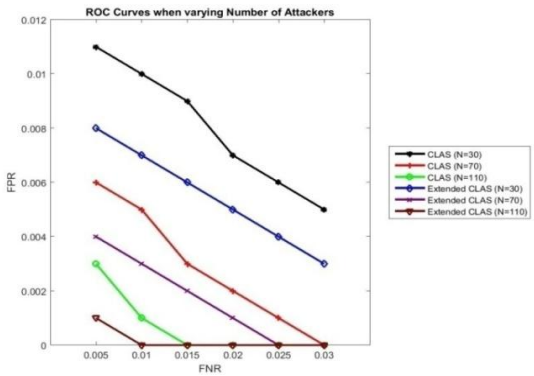

Fig.6: Comparison of ROC when varying Number of Attackers

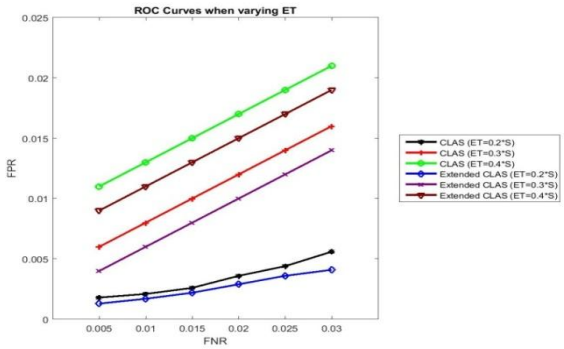

Fig.7: Comparison of ROC when varying ET

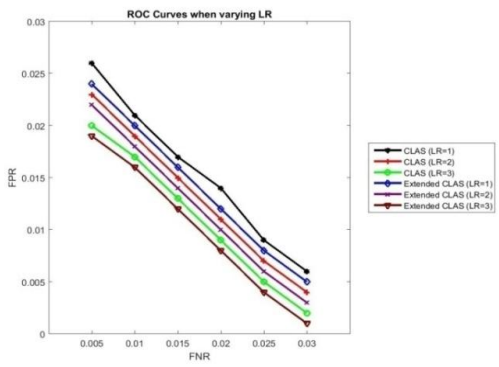

Fig.8: Comparison of ROC when varying LR

The comparison of ROC for extended CLAS and CLAS is shown in Fig. 6, 7 and 8 with varying number of attackers, ET and LR, respectively. In this graph, x-axis denotes the FNR and y-axis denotes the FPR. From this analysis, it is concluded that the proposed extended CLAS achieves the lower FNR and higher FPR when increasing the number of attackers, ET and LR, respectively.

V. CONCLUSION

In this paper, an extended CLAS is proposed for fog-enabled cloud computing networks to authenticate the users efficiently. Based on this proposed extended CLAS, the legitimate login failures due to both selective and arbitrary mobility of users are avoided. Also, the mimic attacks and same location attacks are efficiently defended. Hence, FAR, FNR, login latency overhead, bandwidth and storage overhead are reduced significantly. As a result, this scheme can be very helpful in real-time applications to prevent different attacks and authenticate the legitimate user's login in fog-enabled cloud computing networks.

REFERENCES

1. S. Khan, S. Parkinson and Y. Qin, "Fog computing security: a review of current applications and security solutions", J. Cloud Comput., vol. 6, no. 1, p. 19, 2017.
2. R. Rios, R. Roman, J. A. Onieva and J. Lopez, "From SMOG to Fog: a security perspective", in IEEE Second Int. Conf. Fog Mob. Edge Comput., pp. 56-61, 2017.
3. R. Mahmud, R. Kotagiri and R. Buyya, "Fog computing: a taxonomy, survey and future directions", in Internet of Everything, Springer, Singapore, pp. 103-130, 2018.
4. I. Stojmenovic, S. Wen, X. Huang and H. Luan, "An overview of fog computing and its security issues", Concurr. Comput. Pract. Exp., vol. 28, no. 10, pp. 2991-3005, 2016.
5. H. M. Sun, Y. H. Chen and Y. H. Lin, "oPass: a user authentication protocol resistant to password stealing and password reuse attacks", IEEE Trans. Inf. Forensics Secur., vol. 7, no. 2, pp. 651-663, 2012.
6. Z. Dou, I. Khalil and A. Khreishah, "CLAS: a novel communications latency based authentication scheme", Secur. Commun. Netw., 2017.
7. B. Waters, "Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization", in Int. Workshop Public Key Cryptogr., Springer, Berlin, Heidelberg, pp. 53-70, 2011.
8. M. H. Ibrahim, "Octopus: an edge-fog mutual authentication scheme", Int. J. Netw. Secur., vol. 18, no. 6, pp. 1089-1101, 2016.
9. H. Wang, Z. Wang and J. Domingo-Ferrer, "Anonymous and secure aggregation scheme in fog-based public cloud computing", Future Gener. Comput. Syst., 2017.
10. Y. Jiang, W. Susilo, Y. Mu and F. Guo, "Ciphertext-policy attribute-based encryption against key-delegation abuse in fog computing", Future Gener. Comput. Syst., 2017.
11. A. Alrawais, A. Alhothaily, C. Hu, X. Xing and X. Cheng, "An attribute-based encryption scheme to secure fog communications", IEEE Access, vol. 5, pp. 9131-9138, 2017.
12. R. Yang, Q. Xu, M. H. Au, Z. Yu, H. Wang and L. Zhou, "Position based cryptography with location privacy: a step for fog computing", Future Gener. Comput. Syst., 2017.
13. L. Nazareth and P. Tseng, "Gilding the lily: a variant of the Nelder-Mead algorithm based on golden-section search", Comput. Optim. Appl., vol. 22, no. 1, pp. 133-144, 2002.

AUTHORS PROFILE

C. Nagarani received the M.Sc. Computer Science degree from Periyar University, India in the year 2004 and M.Phil degree in Computer Science from Periyar University, India in the year 2007 respectively. Currently, she is an Assistant Professor of Computer Science, PSG College of Arts and Science, Coimbatore, affiliated to Bharathiyar University. She has a total experience of over 15 years. She has published 4 papers in Journals. Her area of interest is network security.

Dr. R. Kousalya received the B.Sc. degree in Physics from P.S.G.R. Krishnammal College for Women, India in the year 1997, the MCA degree in Computer Applications from Bharathiar University in the year 2000, the M.Phil degree in Computer Science from Manonmaniam Sundaranar University in the year 2003 and the Ph.D degree in Computer Applications from Manonmaniam Sundaranar University in the year 2016. Currently, she is a Head of Department/Professor of Computer Application, Dr. N.G.P Arts and Science College, Coimbatore, affiliated to Bharathiyar University. She has a total experience of over 19 years. She has published 33 papers in Conference and 11 papers in Journal. Her area of interest includes data mining and web mining.