Researchpedia
Open Access Journals

Received: 15 June 2020; **Revised**: 29 October 2021; **Accepted**: 12 December 2021; **Published Online**: 26 December 2021
Researchpedia Journal of Computing, Volume 2, Issue 1, Article 11, Pages 117–126, December 2021

Digital Object Identifier 10.1109/RpJC.2020.DOI Number

A Symmetric Cryptography Based Key Agreement Protocol For Distributed Cloud Computing Environment

Imran Hayat[1], Shehzad Ashraf Chaudhry[2] and Azeem Irshad[1]
[1]Department of Computer Science & Software Engineering, International Islamic University Islamabad, 44000, Pakistan (e-mail: imranhayat974@gmail.com, irshadazeem2@gmail.com)
[2]Department of Computer Engineering, Istanbul Gelisim University Istanbul, Turkey (e-mail:sashraf@gelisim.edu.tr)

Corresponding author: Shehzad Ashraf Chaudhry

ABSTRACT In literature, many researchers put forward different types of authentication schemes for Distributed Cloud Computing environments. Some of the recently proposed schemes are based on a multi-factor authentication system, but such schemes are either vulnerable to different attacks and have large computation times. In 2018 Amin et al. provided a new scheme for Distributed Cloud Computing Environment and claimed that their scheme is much secure. Amin et al. introduced a new notion of perfect anonymity in distributed cloud computing environments where the Control server and cloud server remain unable to recognize the identity of a user requesting to login. Such notion of perfect anonymity is error nous and is not desirable in the distributed cloud computing environment, because if cloud servers and control servers are not able to know the identity of a user, they will not be able to provide user-specific services as per user requirement. We then proposed an improved scheme to combat the incorrectness and to provide the required security.

Keywords Lightweight Authentication, Privacy preserving, Symmetric key, Cloud Computing

I. INTRODUCTION

In the modern era Internet of things (IoT) has become one of the most trendy techniques. Handling of the data, generated from various smart devices in the IoT environment is the most important issue. It is the interconnected things such as sensor tags, devices, and smart objects over the Internet as well. The core focus of IoT is to get information from the environment which can be shared among various devices. It is an important technology of the current modern inter-connected world. [1], [2] Lifestyle of People is improved by using home sensor devices. IoT generally consists of sensors having low memory, network limitations, and low power and battery. So a standard platform is needed which can efficiently handle a large number of heterogeneous devices as well as data that is growing [1] very rapidly. There is no single strategy for realizing the vision of the IoT, as services can be provisioned in various ways.

In a centralized approach, application platforms located on the Internet (e.g. cloud services) acquire information from entities located in data acquisition networks and provide raw data and services to other entities. These application platforms control the whole information flow, and there is little or no support for accessing the information providers directly. Multiple industrial solutions make use of this approach. On the other hand, in a distributed approach, not only the intelligence and the provisioning of services are located at the edge of the network, but also various application platforms can collaborate dynamically. IoT network benefits not one but all i.e. individuals, society, stakeholders of businesses, etc. since IoT network saves time and money. IoT systems deliver faster and accurately with minimum utilization of energy [1]. This improves the quality of life. IoT concept is used in home security devices which are monitored and controlled either locally or remotely using easy-to-use applications available on mobile phones or smartphones. This concept is also used in the office building to handle and

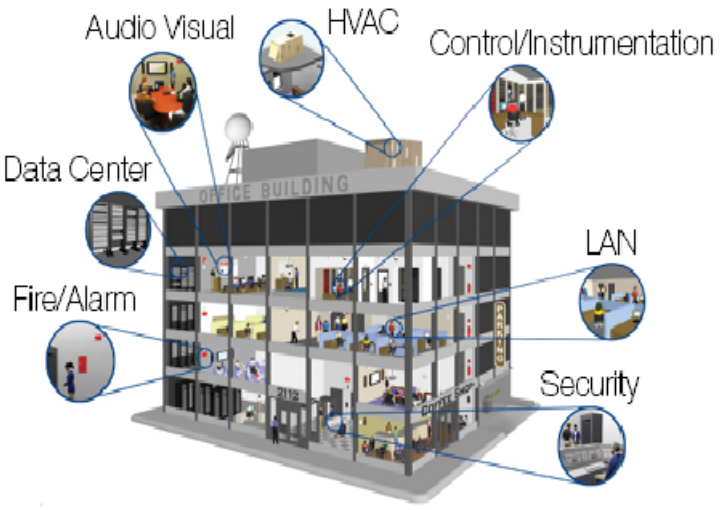

FIGURE 1. IoT Environment

monitor different issues in a modern way as shown in Fig. 1 . Typical IoT devices are security alarms, Cameras, sensors, door locks, etc. are used in a home automation environment. IoT is used in asset and individual tracking, inventory control, energy conservation, shipping, etc. It is used for patient monitoring i.e. various types of wireless sensors are installed on the patient body which communicates with the IoT network and provide all the required information of the patient under treatment. As security concerns, IoT is managed and run by multiple technologies; multiple vendors are involved in it. Due to this fact, privacy is a concern. Security algorithms and certain precautions by the users will help avoid any security-related threats in the IoT network. Security and privacy are the main challenges in managing IoT-based services, particularly in systems including a very large number of devices.

Cloud computing is a powerful technology, by using its information can be accessed from anywhere. All the main issues of IoT can be resolved by using cloud technology. There are many public and private servers currently available, which are providing cloud services. Cloud provides different types of services such as Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). A private cloud can be owned by a single organization and the people of that organization can use the services of that cloud by using the internet through authentication. As there is fast development in Internet and electronic commerce technology, many services are provided to the user/client through Internet communication such as distributed electronic medical records systems, online shopping, and online games, etc. In a cloud environment Cloud security [3] is an important issue among all these applications.

Cloud computing is a paradigm of computing that provides various information technology resources with a high level of scalability using internet technology to a large number of users [2] as shown in Fig. 2. In the cloud computing environment users make access to a large scale computing environment through their computing devices connected to the Internet, use necessary information technology resources including operating systems, platforms, applications, storage, etc. as much as they want and at any time that they want, and pay the fee based upon the number of resources of cloud that they have used. From a hardware provisioning and pricing point of view, a cloud computing environment has many advantages [4]. One advantage is the appearance of unlimited computing resources available on-demand, quickly enough to follow load surges, thereby eliminating the need for cloud computing users to plan far ahead for provisioning. Another advantage is the removal of an up-front commitment by cloud users, thereby allowing companies to start small and increase hardware resources only when there is an increase in the needs of users. The ability to pay for use of computing resources on a short-term basis as needed and release them as needed, thereby rewarding protection by letting storage and machines go when they are no longer useful is also a big advantage of cloud computing. A distributed cloud can reduce costs, communication overheads, and latencies by offering nearby storage resources and computation as well. A distributed cloud connecting geographically distributed, multiple, and smaller data centers, can be an attractive alternative to today's centralized data centers. Better data locality can also improve privacy in distributed cloud computing environment [5]. Authentication is a process of identifying an individual, generally based on a username and password. In insecurity systems, authentication is distinct from authorization, which is the process of giving individuals access to system objects based on their individual given identity. Security and privacy are the main challenges in managing IoT-based services, particularly in systems including a very large number of devices and these security and privacy

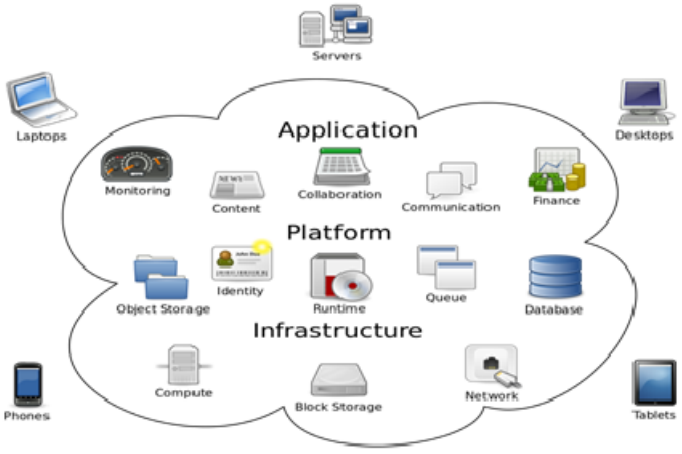

FIGURE 2. Cloud Computing

issues can be resolved through authentication. Authentication protocols [6] have the capability of simply authenticating the connecting party or authenticating the connecting party as well as authenticating itself to the connecting party.

II. LITERATURE REVIEW

In distantly accessed computer systems, a user identifies himself to the system by sending a secret password. In three ways a hacker could learn the user's secret password [6] and then impersonate him when interacting with the system. Firstly by gaining access to the information stored inside the system, e.g., reading the password file of the system. Secondly by intercepting the user's communication with the system, e.g., eavesdropping on the line connecting the user's terminal with the system, or observing the execution of the password checking program. Thirdly by the user's unintended disclosure of his password, e.g., choosing an easily guessed password that can be easily guessed by someone. Li et al. [7] proposed a remote password authentication scheme based on a neural network. In this scheme, the server does not store or maintain passwords. In this scheme server only stores the weights of the classification network. According to this network, the server can authenticate the validity of the login user in real-time. One of the core concepts of this scheme is that it applies to both the multiuser as well as multi-server networks. The users of the system can freely choose their password and the servers are required to keep only the pair user ID and password. The user can login to a variety of servers without repetitive registration with each server of them. The password authentication scheme can prevent the replay attack, the intruder cannot obtain a login password through the open network and replay the password to login to a server. Lin et al. [8] have proposed an efficient remote user authentication scheme. This scheme authenticates the validity of a login user without using any verification table or password file as well. The timestamp technique is used in this scheme to work against the replay attack. This scheme can endure both the modification attack and the replay attack. The major breakthrough of this scheme is that it agrees with multi-user as well as with multi-server networks. The user can log into various servers at the same time without repetitive registrations with all the servers, he/she wants to login. The system can also manage users' privileges by using the service period. When the user's service period expires, the central authority will stop the service for that particular user. Users can freely choose and change their passwords offline which is a significant feature of this scheme. Cao et al. [9]suggested that Lin et al. [8] protocol which authenticates the validity of a login user without using any verification table or password file is not secure against impersonation attack and takes large memory for storage public parameter into the memory of smart card for each user. So Cao et al. put a impersonate attack on the Lin et al. protocol. So as long as a single authentication message of that user is observed, the attack allows an adversary to impersonate any user in the system. Juang et al. [10] introduced a user authentication and key agreement scheme with smart cards. In this scheme, only secure one-way hashing functions and symmetric cryptosystems are used in collaboration with smart cards. This approach can considerably enhance the efficiency and provide much functionality for key agreement and user authentication as well. This scheme generates a session key agreed by the user and the server also. This is a nonce-based scheme that does not have a serious time-synchronization problem. Cheng et al. [11] presented a new password authentication protocol using smart cards for the multi-server architecture. the performance of this protocol is better than Juang's scheme [12]. Furthermore, this scheme achieves the essential requirements which include choosing and changing the password at will,

Lower computation, Security, Mutual authentication, Single registration, and Session key agreement, which are regarded as the important criteria of the password authentication protocols. The security of this scheme is also based on the public one-way hash function and symmetric encryption and decryption function, so the scheme is more efficient than Juang's scheme. This scheme can significantly improve the efficiency of the multi-server password authentication protocol such that it can be useful for the real world as well. Liao et al. [13] suggested a key agreement protocol in which the concept of dynamic identity for the multi-server environment based on cryptographic hash function is used. They declared that their protocol satisfies all the important security aspects of a multi-server environment which includes, firstly a secure password change method to prevent the adversary from updating password freely, secondly resistance against various attacks which also includes two-factor security, thirdly the computation cost is more efficient, fourthly it is a nonce-based scheme to avoid the time synchronization problem as well. Hsiang et al. [14] demonstrated that the protocol in [13] is vulnerable to insider's attack, server spoofing attack, registration center spoofing attack, masquerade attacks. and they designed an extended protocol to overcome such types of attacks. They declare that the extended protocol takes low complexity, higher security so its efficiency is better than previous research. Furthermore, the scheme avoids the adversary to breach the secret key from the stolen smartcard or intercept the information when the smartcard was carelessly lost. Sood et al. [15] claimed and criticized that the protocol in [14] is susceptible to replay attack, impersonation attack, and stolen smart card attack, and the password change process is also not accurate in it. So they specified a secure dynamic identity-based authentication protocol for multi-server architecture using smart cards. This is very effective to overcome different attacks as discussed earlier. The protocol helps the service provider servers and the control server to recognize the user by computing individual static identity and at the same time keeps the user's identity dynamic in the communication channel. Li et al. [16] claimed that they developed a countermeasure protocol against the protocol in [15], which is incorrect and does not defend the attack. To improve security, Xue et al. [17] stated that the protocol in [16] is worthless due to not protecting numerous security. Xue et al. developed an improved dynamic pseudonym identity-based authentication and key agreement protocol, which is suitable for the multi-server environment. Compared with related protocols, this protocol is demonstrated to satisfy all the necessary security requirements. Xue et al. [17] presented an improved dynamic pseudonym identity-based authentication and key agreement protocol which comprises on five phases which include; Registration phase, Login phase, Authentication and key agreement phase(having five steps), Password update phase, and identity update phase. Protocol satisfies all the essential security requirements for authentication and key agreement in the multi-server environment. As compared to Li et al.'s protocol and Sood et al.'s protocol, this protocol keeps efficiency. Chuang et al. [18] suggested a secure remote user authentication scheme that not only supports the multi-server environment to reduce the overhead of the RC but also possesses high-security properties to protect the valid user against attacks with minimal computational cost as well. This scheme is suitable for real-life applications because it is a true lightweight authentication scheme that only uses the hash function. This scheme also satisfies the many security properties which include anonymity, no verification tables, mutual authentication, resistance to forgery attacks, no clock synchronization problem, resistance to modification attacks, resistance to replay attacks, fast error detection, resistance to off-line guessing attacks, resistance to insider attacks, simple and secure choice and change of passwords, biometric template protection, and session key agreement as well. Amin et al. [19] criticized that Xue et al. [17] scheme is not fulfilled user anonymity and does not have the resistance against offline-guessing attack, privileged insider attack at the server end, session key disclosure attack, user impersonate attack and also have a design flaw in the authentication phase. Amin et al. [19] also analyzed that Chuang et al. [18] scheme is not provided security against user impersonate attacks.

A. PROBLEM STATEMENT

Recently, many researchers put forward different types of authentication schemes for Distributed Cloud Computing environments. Some of these schemes are based on a multi-factor authentication system, but many of such schemes are either vulnerable to different attacks or have large computation time. In 2016, Amin et al. [19] provided a new scheme for Distributed Cloud Computing Environment, and claimed that their scheme is much secure. In with proposal, we have found that Amin et al. scheme is incorrect.

B. ATTACK MODEL

We adopted the common CK model [20], which is adopted in many authoritative works [21]–[28]. As per the CK model, the attacker in addition to listening to the system can block a message, modify a legal or create a fake message and send it to any of the participants. Moreover, through power analysis, the attacker can extract the parameters stored in the memory of a stolen/captured device.

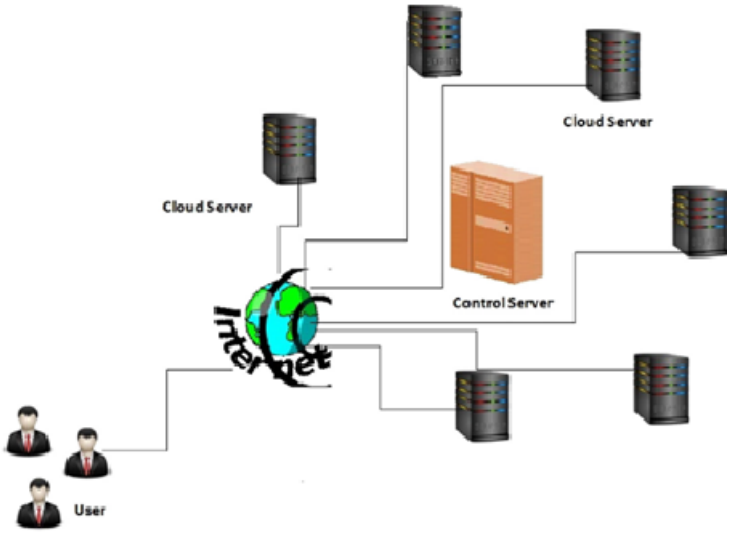

FIGURE 3. Cloud Architecture

III. REVIEW OF AMIN ET AL. SCHEME

A private cloud server stores secret information from the environment using the core concept of the Internet of Things (IoT). To access stored confidential information from the private cloud is an issue. To resolve this issue, Amin et al. [19] protocol put forwards a smartcard-based authentication protocol for distributed cloud environment, where the registered user or client can securely access desired private cloud server. This protocol consists of the phases; (i) Registration Phase (a. Cloud Server registration, b. User Registration), (ii) Login phase, (iii) Authentication phase, (iv) password change phase, and (v) Identity update phase. In this cloud architecture, there are several private cloud servers which are controlled by the control server and all the private cloud servers are located in a distributed manner as shown in Fig. 3. On executing this protocol, a valid user or client can access all the private cloud servers securely. Please refer to the notation guide in Table 1 for a better understanding of the concepts used in this paper.

A. REGISTRATION PHASE

This phase comprises of two sub-phases: i.e. (1) cloud server registration and (2) user registration.

1) Cloud server registration

During cloud server registration, the S_m chooses an identity SID_m, a random number d and sends (SID_m, d) to CS. After receiving it, the CS computes $PSID_m = h(SID_m||d)$, $BS_m = h(PSID_m||y)$ and sends BS_m to S_m securely. Finally, the Sm stores secret parameter (BS_m, d) into his/her memory.

2) User registration phase

During registration in CS, user first chooses desired identity ID_i, password P_i and two random numbers (b_1, b_2). Then, the ID_i computes $A_i = h(P_i||b_1)$, $PID_i = h(ID_i||b_2)$, $bb_i = b_2 \oplus A_i$ and sends (A_i, PID_i) to the CS securely. On getting (A_i, PID_i), the CS calculates: $C_i = h(A_i||PID_i)$, $D_i = h(PID_i||x)$ and $E_i = D_i \oplus A_i$. Finally, the CS prepares and delivers a new smartcard for each U_i after recording $(C_i, E_i, h(\cdot))$ in the smartcard and transports it to U_i through private communication. After getting it, the U_i records (DP, bb_i) in the smartcard, where $DP = h(ID_i||P_i) \oplus b_1$. Finally, the smartcard holds $(C_i, E_i, bb_i, DP, h(\cdot))$. In registration phase two random numbers <b_1, b_2> are used for resisting insider attack.

TABLE 1. Notations table

Symbol	Description
CR	Card Reader
S_j	j^{th} service provider server
S_m	m^{th} cloud server
U_i	i^{th} user
CS	The control server
ID_i	Identity of the userU_i
P_i	Password of the userU_i
x	Secret number only known to CS
y	Secret number of CS
d	Random number of S_j
b	Random number of U_i
h(..)	Hash function $(0,1)^l \rightarrow (0,1)^n$
T	Timestamp
ΔT	Estimated time delay
SK	Secret session key
$\oplus$	Bit-wise xor operation
$\|\|$	Concatenate operation

B. LOGIN PHASE

For accessing server resources, a legal user U_i first punches the smartcard into card reader CR and inputs ID_i and P_i to the terminal. Then, the card reader calculates ${b^*}_1 = DP \oplus h(ID_i||P_i)$, ${A_i}^* = h(P_i||b_1)$, ${b_2}^* = bb_i \oplus A_i$, ${PID_i}^* = h(ID_i||b_2)$, ${C_i}^* = h(A_i||PID_i)$. Then, the CR checks the condition $({C_i}^*? = C_i)$. If $({C_i}^* == C_i)$, it means that $({ID_i}^* = ID_i)$ and $({P_i}^* = P_i)$. The CR produces a 128 bit random number N_i and computes the following operations : $D_i = E_i \oplus A_i$, $G_i = h(PID_i||SID_m||N_i||TS_i||D_i)$, $F_i = D_i \oplus N_i$, $Z_i = SID_m \oplus h(D_i||N_i)$, where SID_m is the cloud server's identity chosen by the user U_i. Then, the CR transmits the login messages $(G_i, F_i, Z_i, PID_i, TS_i)$ to the S_m publicly.

1) Authentication phase

This phase is necessary for performing mutual authentication as well as key agreement among U_i, S_m and CS. The details explanation of this phase are as follows.

Step 1:The S_m first checks the condition whether $(TS_m - TS_i < \Delta T)$ holds or not on receiving the login message, where $TS_m, \Delta T$ are the cloud server's current timestamp and expected valid time interval for transmission delay respectively. If the condition is not true, the S_m terminates the connection; otherwise, the S_m produces a 128 bit random number N_m and computes the operations: $J_i = BS_m \oplus N_m$, $K_i = h(N_m||BS_m||G_i||TS_j)$. Finally, the S_m sends $(J_i, K_i, PSID_m, G_i, F_i, Z_i, PID_i, TS_i, TS_m)$ to the CS publicly.

Step 2:On getting messages from S_m, CS first checks the time interval i.e. $(TS_{cs} - TS^m < \Delta T^*)$, where TS_{cs}, ΔT are the CS's current timestamp and expected valid time interval for transmission delay respectively. If the verification holds, CS executes the following operations; otherwise, terminates the session. $D_i = h(PID_i||x)$, ${N_i}^* = F_i \oplus D_i$, ${SID_m}^* = Z_i \oplus h(D_i||{N_i}^*)$, ${G_i}^* = h(PID_i||{SID_m}^*||{N_i}^*||D_i||TS_i)$. After that, the CS checks the condition $({G_i}^*? = G_i)$. If $({G_i}^*? = G_i)$, the CS thinks that the U_i is legal; otherwise, terminates the procedures. After that, the CS computes the operations:${BS_m}^* = h(PSID_m||y)$, ${N_m}^* = {BS_m}^* \oplus J_i$, ${K_i}^* = h({BS_m}^*||{N_m}^*||G_i||TS_m)$. Again, the CS checks the condition $({K_i}^*? = K_i)$. If $({K_i}^*? = K_i)$, the CS thinks that S_m is legal; otherwise, terminates the procedure. After that, the CS chooses a 128 bit random number N_{cs} and computes the operations: $P_{cs} = N_m \oplus N_{cs} \oplus h(Ni||Di)$, $R_{cs} = N_i \oplus N_{cs} \oplus h({BS_m}^*||{N_m}^*)$, $SK_{cs} = h(N_i \oplus N_m \oplus N_{cs})$, $Q_{cs} = h((N_m \oplus N_{cs})||SK_{cs})$, $V_{cs} = h((N_i||N_{cs})||SK_{cs})$. where SK_{cs} is the secret session key. Finally, the CS sends $(P_{cs}, R_{cs}, Q_{cs}, V_{cs})$ to the S_m for achieving mutual authentication of the protocol through public communication.

Step 3:On getting reply messages from CS, the S_m computes the operations: $W_m = h(BS_m||N_m)$, $N_i \oplus N_{cs} = R_{cs} \oplus W_m$, $SK_m = h(N_i \oplus N_{cs} \oplus N_m)$, $V_{cs} = h((N_i \oplus N_{cs})||SK_m)$. Then, the S_m checks the condition $({V_{cs}}^*? = V_{cs})$ or not. If $({V_{cs}}^* \neq V_{cs})$, terminates the session; otherwise, sends messages (P_{cs}, Q_{cs}) to the U_i publicly.

Step 4:On obtaining messages from S_m, the U_i calculates the operations: $L_i = h(Ni||D_i)$, $N_m \oplus N_{cs} = P_{cs} \oplus L_i$, $SK_i = h(N_m \oplus N_{cs} \oplus N_i)$, ${Q_{CS}}^* = h((N_m \oplus N_{cs})||SK_i)$. Then, the U_i checks the condition $({Q_{CS}}^*? = Q_{CS})$ and if $({Q_{CS}}^* == Q_{CS})$, it proves the authenticity of S_m and CS. Finally, the proposed protocol achieves mutual authentication among U_i, S_m and CS.Now, the U_i and the S_m can exchange their secret information securely using $SK_m = SK_i$.

2) Password change phase

Whenever an existing U_i wishes to renew password, first he/she provides ID_i and P_i after punching the smartcard. Then, the CR executes the operations:$b_1 = DP \oplus h({ID_i}^*||{P_i}^*)$, $A_i = h({P_i}^{new}||b_1)$, $b_1 = bb_i \oplus A_i$, $PID_i = h({ID_i}^*||b_2)$, , $C_i = h({A_i}^*||{PID_i}^*)$. The smartcard checks the condition $(C_i^*? = C_i)$. If $(C_i^* == C_i)$, the card reader requests to enter a new password ${P_i}^{new}$ to the U_i and calculates the operations: ${A_i}^{new} = h({P_i}^{new}||b_1)$, ${C_i}^{new} = h({A_i}^{new}||{PID_i}^{new})$, $D_i = E_i \oplus A_i = h({PID_i}^{new}||x)$, $bb_i = {b_2}^* \oplus {A_i}^{new}$, ${E_i}^{new} = D_i \oplus {A_i}^{new}$, $DP^{new} = h(ID_i||{P_i}^{new}) \oplus {b_1}^*$. Finally, the CR substitutes $({C_i}^{new}, {E_i}^{new}, bb_i, DP^{new})$ in the place of $(C_i, E_i, bb_i, DP^{new})$ respectively in the smartcard. Thus, a user can renew password without facing any difficulty.

IV. CRYPTANALYSIS OF AMIN ET AL. SCHEME

Cryptanalysis of Amin et al. [19] scheme is performed in this section. It is found that the Identity Update Phase is useless and does not make sense in the practical application of the scheme because no one wants to change his/her identity once adopted for further communication. Once the identity has changed there is no way for others to identify that one with a new identity. During analysis of the login and authentication phase of this scheme, it is found that during authentication it does not reveal the user identity ID_i to S_m and CS. As in this scheme CR sends $(G_i; F_i; Z_i; PID_i; TS_i)$ to S_m then S_m sends $(J_i; K_i; PSID_m; G_i; F_i; Z_i; PID_i; TS_i; TS_m)$ to the CS which is not correct. The analysis shows that the scheme has the incorrect notion of Perfect Anonymity.

A. INCORRECT NOTION OF PERFECT ANONYMITY

Amin et al. introduced a new notion of perfect anonymity in distributed cloud computed environment where a CS controls all the Cloud Servers S_m, where Control server CS and Clouds S_m remain unable to recognize the identity of the user requesting to login. In our view, such notion of perfect anonymity is error nous and is not desirable in the distributed cloud computing environment, because if S_m and CS are not able to know the identity of a user, they will not be able to provide user-specific services as per user requirement.

V. PROPOSED SOLUTION

We briefly explain our proposed solution through Fig. 4 and in following subsections:

A. REGISTRATION PHASE

This phase comprises two subsections i.e. (1) cloud server registration and (2) user registration, explained as follows:

1) Cloud server registration

During cloud server registration, the S_m chooses an identity SID_m, a random number d and sends (SID_m, d) to CS. After receiving it, the CS computes $PSID_m = h(SID_m||d)$, $BS_m = h(PSID_m||y)$ and sends BS_m to S_m securely. Finally, the S_m stores secret parameter (BS_m, d) into his/her memory.

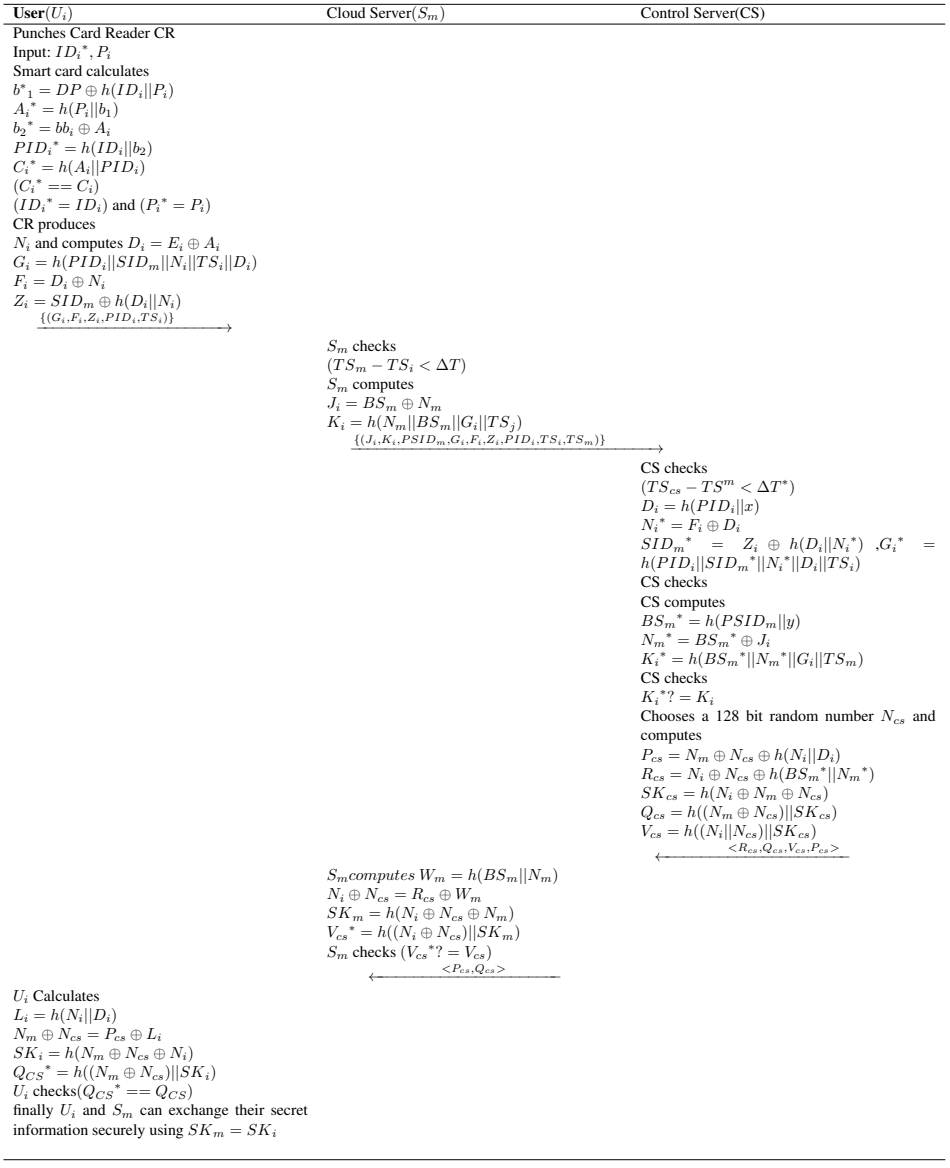

FIGURE 4. Login and Authentication

2) User registration phase

During registration in CS, user first chooses desired identity ID_i, password P_i and two random numbers (b_1, b_2). Then, the ID_i computes $A_i = h(P_i||b_1)$, $PID_i = h(ID_i||b_2)$, $bb_i = b_2 \oplus A_i$ and sends (A_i, PID_i) to the CS securely. On getting (A_i, PID_i), the CS calculates: $C_i = h(A_i||PID_i)$, $D_i = h(PID_i||x)$ and $E_i = D_i \oplus A_i$. Finally, the CS prepares and delivers a new smartcard for each U_i after recording $(C_i, E_i, h(\cdot))$ in the smartcard and transports it to U_i through private communication. After getting it, the U_i records (DP, bb_i) in the smartcard, where $DP = h(ID_i||P_i) \oplus b_1$. Finally, the smartcard holds $(C_i, E_i, bb_i, DP, h(\cdot))$. In registration phase two random numbers <b_1, b_2> are used for resisting insider attack.

B. LOGIN AND AUTHENTICATION PHASE

For accessing server resources, a legal user U_i first punches the smartcard into card reader CR and inputs ID_i and P_i to the terminal. Then, the card reader calculates $b^*{}_1 = DP \oplus h(ID_i||P_i)$, $A_i{}^* = h(P_i||b_1)$, $b_2{}^* = bb_i \oplus A_i$, $PID_i{}^* = h(ID_i||b_2)$, $C_i{}^* = h(A_i||PID_i)$.Then, the CR checks the condition $(C_i{}^*? = C_i)$. If $(C_i{}^* == C_i)$, it means that $(ID_i{}^* = ID_i)$ and $(P_i{}^* = P_i)$. The CR produces a 128 bit random number N_i and computes the following operations : $D_i = E_i \oplus A_i$, $G_i = h(PID_i||SID_m||N_i||TS_i||D_i)$, $F_i = D_i \oplus N_i$, $Z_i = SID_m \oplus h(D_i||N_i)$. where SID_m is the cloud server's identity chosen

by the user U_i. Then, the CR transmits the login messages $(G_i, F_i, Z_i, PID_i, TS_i)$ to the S_m publicly. Authentication phase is necessary for performing mutual authentication as well as key agreement among U_i, S_m and CS. The details explanation of this phase are as follows:

Step 1:The S_m first checks the condition whether $(TS_m - TS_i < \Delta T)$ holds or not on receiving the login message, where $TS_m, \Delta T$ are the cloud server's current timestamp and expected valid time interval for transmission delay respectively. If the condition is not true, the S_m terminates the connection; otherwise, the S_m produces a 128 bit random number N_m and computes the operations: $J_i = BS_m \oplus N_m$, $K_i = h(N_m||BS_m||G_i||TS_j)$. Finally, the S_m sends $(J_i, K_i, PSID_m, G_i, F_i, Z_i, PID_i, TS_i, TS_m)$ to the CS publicly.

Step 2:On getting messages from S_m, CS first checks the time interval i.e. $(TS_{cs} - TS^m < \Delta T^*)$, where $TS_{cs}, \Delta T$ are the CS's current timestamp and expected valid time interval for transmission delay respectively. If the verification holds, CS executes the following operations; otherwise, terminates the session. $D_i = h(PID_i||x)$, ${N_i}^* = F_i \oplus D_i$, ${SID_m}^* = Z_i \oplus h(D_i||{N_i}^*)$, ${G_i}^* = h(PID_i||{SID_m}^*||{N_i}^*||D_i||TS_i)$.
After that, the CS checks the condition $({G_i}^*? = G_i)$. If $({G_i}^*? = G_i)$, the CS thinks that the U_i is legal; otherwise, terminates the procedures. After that, the CS computes the operations:${BS_m}^* = h(PSID_m||y)$, ${N_m}^* = {BS_m}^* \oplus J_i$, ${K_i}^* = h({BS_m}^*||{N_m}^*||G_i||TS_m)$. Again, the CS checks the condition $({K_i}^*? = K_i)$. If $({K_i}^*? = K_i)$, the CS thinks that S_m is legal; otherwise, terminates the procedure.
After that, the CS chooses a 128 bit random number N_{cs} and computes the operations: $P_{cs} = N_m \oplus N_{cs} \oplus h(Ni||Di)$, $R_{cs} = N_i \oplus N_{cs} \oplus h({BS_m}^*||{N_m}^*)$, $SK_{cs} = h(N_i \oplus N_m \oplus N_{cs})$, $Q_{cs} = h((N_m \oplus N_{cs})||SK_{cs})$, $V_{cs} = h((N_i||N_{cs})||SK_{cs})$. where SK_{cs} is the secret session key. Finally, the CS sends $(P_{cs}, R_{cs}, Q_{cs}, V_{cs})$ to the S_m for achieving mutual authentication of the protocol through public communication.

Step 3:On getting reply messages from CS, the S_m computes the operations: $W_m = h(BS_m||N_m)$, $N_i \oplus N_{cs} = R_{cs} \oplus W_m$, $SK_m = h(N_i \oplus N_{cs} \oplus N_m)$, $V_{cs} = h((N_i \oplus N_{cs})||SK_m)$. Then, the S_m checks the condition $({V_{cs}}^*? = V_{cs})$ or not. If $({V_{cs}}^* \neq V_{cs})$, terminates the session; otherwise, sends messages (P_{cs}, Q_{cs}) to the U_i publicly.

Step 4:On obtaining messages from S_m, the U_i calculates the operations: $L_i = h(Ni||D_i)$, $N_m \oplus N_{cs} = P_{cs} \oplus L_i$, $SK_i = h(N_m \oplus N_{cs} \oplus N_i)$, ${Q_{CS}}^* = h((N_m \oplus N_{cs})||SK_i)$. Then, the U_i checks the condition $({Q_{CS}}^*? = Q_{CS})$ and if $({Q_{CS}}^* == Q_{CS})$, it proves the authenticity of S_m and CS. Finally, the proposed protocol achieves mutual authentication among U_i, S_m and CS. Now, the U_i and the S_m can exchange their secret information securely using $SK_m = SK_i$.

C. PASSWORD CHANGE PHASE

Whenever an existing U_i wishes to renew password, first he/she provides ID_i and P_i after punching the smartcard. Then, the CR executes the operations:$b_1 = DP \oplus h({ID_i}^*||{P_i}^*)$, $A_i = h({P_i}^{new}||b_1)$, $b_1 = bb_i \oplus A_i$, $PID_i = h({ID_i}^*||b_2)$, , $C_i = h({A_i}^*||{PID_i}^*)$. The smartcard checks the condition $({C_i}^*? = C_i)$. If $({C_i}^* == C_i)$, the card reader requests to enter a new password ${P_i}^{new}$ to the U_i and calculates the operations: ${A_i}^{new} = h({P_i}^{new}||b_1)$, ${C_i}^{new} = h({A_i}^{new}||{PID_i}^{new})$, $D_i = E_i \oplus A_i = h({PID_i}^{new}||x)$, , ${bb_i} = {b_2}^* \oplus {A_i}^{new}$, ${E_i}^{new} = D_i \oplus {A_i}^{new}$, $DP^{new} = h(ID_i||{P_i}^{new}) \oplus {b_1}^*$. Finally, the CR substitutes $({C_i}^{new}, {E_i}^{new}, bb_i, DP^{new})$ in the place of $(C_i, E_i, bb_i, DP^{new})$ respectively in the smartcard. Thus, a user can renew password without facing any difficulty.

VI. CONCLUSION

In this paper, we first reviewed and analyzed a recent authentication scheme designed specifically for distributed cloud computing environments by Amin et al. It is also claimed that Amin et al.'s scheme provides perfect anonymity. We have shown in this paper that the scheme proposed by Amin et al. cannot work in distributed cloud computing environments due to the design fault that occurred cause of the incorrect notion of perfect forward anonymity. We then proposed the improved scheme to work in distributed cloud computing environments. The proposed scheme provides security as well as anonymity and can resist the known attack.

REFERENCES

[1] A. Zaslavsky, C. Perera, D. Georgakopoulos, Sensing as a service and big data, arXiv preprint arXiv:1301.0159 (2013).

[2] A. Irshad, M. Sher, H. F. Ahmad, B. A. Alzahrani, S. A. Chaudhry, R. Kumar, An improved multi-server authentication scheme for distributed mobile cloud computing services, KSII Transactions on Internet and Information Systems (TIIS) 10 (12) (2016) 5529–5552.

[3] Chang, Victor and Kuo, Yen-Hung and Ramachandran, Muthu, Cloud computing adoption framework: A security framework for business clouds, Future Generation Computer Systems 57 (2016) 24–41.
[4] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al., A view of cloud computing, Communications of the ACM 53 (4) (2010) 50–58.
[5] Y. Coady, O. Hohlfeld, J. Kempf, R. McGeer, S. Schmid, Distributed cloud computing: Applications, status quo, and challenges, ACM SIGCOMM Computer Communication Review 45 (2) (2015) 38–43.
[6] Duncan, Richard, An overview of different authentication methods and protocols, Report submitted to SANS Institute (2001).
[7] Li, Li-Hua and Lin, Luon-Chang and Hwang, Min-Shiang, A remote password authentication scheme for multiserver architecture using neural networks, IEEE Transactions on Neural Networks 12 (6) (2001) 1498–1504.
[8] Lin, Iuon-Chang and Hwang, Min-Shiang and Li, Li-Hua, A new remote user authentication scheme for multi-server architecture, Future Generation Computer Systems 19 (1) (2003) 13–22.
[9] Cao, Xiang and Zhong, Sheng, Breaking a remote user authentication scheme for multi-server architecture, IEEE Communications Letters 10 (8) (2006) 580–581.
[10] Juang, Wen-Shenq, Efficient password authenticated key agreement using smart cards, Computers & Security 23 (2) (2004) 167–173.
[11] Chang, Chin-Chen and Lee, Jung-San, An efficient and secure multi-server password authentication scheme using smart cards, in: Cyberworlds, 2004 International Conference on, IEEE, 2004, pp. 417–422.
[12] W.-S. Juang, Efficient password authenticated key agreement using smart cards, Computers & Security 23 (2) (2004) 167–173.
[13] Liao, Yi-Pin and Wang, Shuenn-Shyang, A secure dynamic id based remote user authentication scheme for multi-server environment, Computer Standards & Interfaces 31 (1) (2009) 24–29.
[14] Hsiang, Han-Cheng and Shih, Wei-Kuan, Improvement of the secure dynamic id based remote user authentication scheme for multi-server environment, Computer Standards & Interfaces 31 (6) (2009) 1118–1123.
[15] Sood, Sandeep K and Sarje, Anil K and Singh, Kuldip, A secure dynamic identity based authentication protocol for multi-server architecture, Journal of Network and Computer Applications 34 (2) (2011) 609–618.
[16] Li, Xiong and Xiong, Yongping and Ma, Jian and Wang, Wendong, An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards, Journal of Network and Computer Applications 35 (2) (2012) 763–769.
[17] Xue, Kaiping and Hong, Peilin and Ma, Changsha, A lightweight dynamic pseudonym identity based authentication and key agreement protocol without verification tables for multi-server architecture, Journal of Computer and System Sciences 80 (1) (2014) 195–206.
[18] Chuang, Ming-Chin and Chen, Meng Chang, An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics, Expert Systems with Applications 41 (4) (2014) 1411–1418.
[19] R. Amin, N. Kumar, G. Biswas, R. Iqbal, V. Chang, A light weight authentication protocol for iot-enabled devices in distributed cloud computing environment, Future Generation Computer Systems 78 (2018) 1005–1019.
[20] R. Canetti, H. Krawczyk, Analysis of key-exchange protocols and their use for building secure channels, in: International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2001, pp. 453–474.
[21] K. Mahmood, J. Arshad, S. A. Chaudhry, S. Kumari, An enhanced anonymous identity-based key agreement protocol for smart grid advanced metering infrastructure, International Journal of Communication Systems 32 (16) (2019) e4137.
[22] T.-Y. Wu, L. Yang, Q. Meng, X. Guo, C.-M. Chen, Fog-driven secure authentication and key exchange scheme for wearable health monitoring system, Security and Communication Networks 2021 (2021) 8368646. doi:10.1155/2021/8368646.
[23] S. A. Chaudhry, K. Yahya, M. Karuppiah, R. Kharel, A. K. Bashir, Y. B. Zikria, Gcacs-iod: A certificate based generic access control scheme for internet of drones, Computer Networks 191 (2021) 107999. doi:10.1016/j.comnet.2021.107999.
[24] Y. Zhang, D. He, L. Li, B. Chen, A lightweight authentication and key agreement scheme for internet of drones, Computer Communications 154 (2020) 455–464.
[25] S. A. Chaudhry, ombating identity de-synchronization: An improved lightweight symmetric key based authentication scheme for iov, Journal of Network Intelligence 6 (2021) 656–667.
[26] M. N. Aman, M. H. Basheer, B. Sikdar, Data provenance for iot with light weight authentication and privacy preservation, IEEE Internet of Things Journal 6 (6) (2019) 10441–10457.
[27] J. Srinivas, A. K. Das, N. Kumar, J. J. P. C. Rodrigues, Tcalas: Temporal credential-based anonymous lightweight authentication scheme for internet of drones environment, IEEE Transactions on Vehicular Technology 68 (7) (2019) 6903–6916.
[28] M. A. Saleem, S. H. Islam, S. Ahmed, K. Mahmood, M. Hussain, Provably secure biometric-based client–server secure communication over unreliable networks, Journal of Information Security and Applications 58 (2021) 102769.