Journal of Applied Research and Technology

CCADET
CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLÓGICO

www.jart.ccadet.unam.mx
Journal of Applied Research and Technology 16 (2018) 9-21

Original

An efficient 3D Diffie-Hellman based Two-Server password-only authenticated key exchange

Anitha Kumari K. [a,*]
Sudha Sadasivam G. [b]

[a]Department of IT, PSG College of Technology, India
[b]Department of CSE PSG College of Technology, India

Received dd mm aaaa; accepted dd mm aaaa
Available online dd mm aaaa

Abstract: In emerging technological world, security potentially remains as a highest challenge in the large-scale distributed systems, as it is suffering extensively with adversarial attacks due to insufficient mutual authentication. In order to address this, a state-of-art tetrahedron (3D) based two-server Password Authenticated and Key Exchange (PAKE) protocol has been formulated with formal proof of security by incorporating the elementary properties of plane geometry. The main intention of this work is, obtaining a password from the stored credentials must be infeasible when both the servers compromised together. At the outset to realize these goals, in this paper, the properties of the tetrahedron are utilized along with Diffie-Hellman (DH) key exchange algorithm to withstand against malicious attacks. A significant aspect of the proposed 3D PAKE protocol is, client side complexity has been reduced to a greater extent in terms of computation and communication. Both theoretically and practically, 3D PAKE protocol is the first demonstrable secure two-server PAKE protocol that breaks the assumptions of the Yang et al. and Yi et al. protocol that the two servers must not compromise together. Computational complexity, communication complexity, security key principles, best of all attacks happening dubiously are considered as the evaluation parameters to compare the performance of the proposed 3D PAKE protocol.

Keywords: 3D PAKE protocol; tetrahedron property analysis; Diffie-Hellman key exchange.

1. INTRODUCTION

In this digital world, web services are accessed by the users consistently. Yet, these web services are suffering with poor authentication that in turn allows malicious users to impersonate the services. Thereby, framing an effective solution to reduce the attack surface is inevitable. Most of the web services rely upon digital certificate for verification. On the contrary, when the certificate authority is vulnerable to hazardous attacks or security breaches, the primary effect includes compromising of numerous certificates (Dennis, 2012). An optimal and effectual solution to address this issue is PAKE protocol. PAKE establishes a secret key between two communicating parties based upon the knowledge of sensitive information like, low-entropy password (Bellovin & Merritt, 1992). Relatively, password based authentication techniques is a flexible one to reduce the intricacies to a greater extent without demanding abundant space or device requirement. It is considered as one of the simplest and most convenient authentication mechanisms. In PAKE, an attacker or man-in-the-middle will not be able to guess a password without further interactions with communicating parties. This defensive

[*] Corresponding author.
E-mail address: anitha.psgsoft@gmail.com (Anitha Kumari K.).

Peer Review under the responsibility of Universidad Nacional Autónoma de México.

http://

property acts as a phenomenal aspect of PAKE. In most of the cases, the single server model is liable to invasive attacks, whereas the multi - server model is expensive and entails high communication bandwidth. With that note, two-server model is considered as a wise choice. 3D protocol assures that determining the key/obtaining a password from the stored information is impossible by the adversaries.

Mathematical research normally simplifies a complex problem in all academic disciplines. Using geometrical properties in a PAKE protocol is an appealing technique, as this plays an extensive role in real life from the most basic to the advanced part. An amazing fact is retrieving the original source from these properties is infeasible (Jack, 2008).The security model of the proposed protocol is based on the properties - circumcenter (ω) and the angle between the medians (θ) of tetrahedron to protect the system against attacks. It is a proven fact, that the properties of a tetrahedron are undoubtedly more difficult to visualize and break (Choate, 1976). A profound analysis of the protocol acts as an evidence for the protocol's resistance against the attacks.
Introduction section addresses the motivation of choosing trigonometric properties in 3D PAKE protocol. Section 2 explores the related literature, section 3 elaborates the proposed methodology, section 4 converses the protocols' correctness and security analysis theoretically, section 5 carries out performance analysis and the section 6 presents the summary of the key contributions of 3D PAKE and possible research avenues.

2. RELATED WORK

In web services, Kerberos based framework generates tickets for binary authentication. One of the major limitations of Kerberos is that, it is vulnerable to password guessing attacks (Bellovin & Merritt, 1990). Further, Kerberos requires a trusted path to handle passwords and does not support multipart authentication. The flaws can be inherently resolved by using a formal PAKE protocol.

Initially a pioneering symmetric two-server PAKE is proposed by Katz, MacKenzie, Taban, and Gligor (2005). Computation and communication complexity is the highest barrier in adopting Katz protocol. Three-party encrypted key exchange scheme proposed by Lin, Sun, and Hwang (2000) is stringent against attacks; however, as a prerequisite, the client needs to obtain and verify the the public key of the server. Similarly, computational complexity is the limitation of the nPAKE+ scheme (Wan, Deng, Bao, & Preneel, 2007). A Gateway based Threshold Password-based Authenticated Key Exchange (GTPAKE) scheme is susceptible to undetectable on-line password guessing attack by a malicious gateway (Byun, Lee, & Lim, 2006; Chien, Wu, & Yeh, 2013). A threshold PAKE verifies the client based on the threshold value (Abdalla, Chevassut, & Fouque, 2005; Mackenzie, Shrimpton, & Jakobsson, 2002). Even though, the protocol is secure against dictionary attacks, fixing the acceptable threshold value is a complicated process. 3D password authentication system constituting of recognition, recall, tokens and biometrics as a single authentication system is proposed by Pooja, Shilpi, Sujata, & Vinita, (2012). Device requirement is a limitation of this approach. An efficient password based two-server authentication and pre-shared key exchange system using smart card is proposed by Chouksey & Pandey (2013). It is an ID-based remote user authentication protocol with a smart card that uses simple bitwise XOR operations and hash functions. Device requirement is the main shortcoming of this approach. Yang, Deng, and Bao (2006) proposed the practical two-server PAKE model. It is not robust against dictionary attacks caused by the active adversary and it is possible to compute the session key established between the User (U) and Service Server (SS). Lee and Lee (2007) presented a two-server authentication and key exchange protocol that uses multiple SS with a single Control Server (CS). This protocol is not efficient when compared with Yang et al. (2006) protocol in terms of computational cost. An enhancement of Yang et al. (2006) scheme is proposed by Jin, Wong, & Xu, (2007) as Password-only Two-Server Authenticated Key Exchange (PTAKE) to remain secure against offline dictionary attack. Yet the formal security model has not been devised for PTAKE. An efficient two-server PAKE proposed by Yi, Ling, and Wang (2013) is a symmetric two-server PAKE protocol that performs the operations in parallel at both the servers. However, for transferring messages it relies upon a gateway that is expensive and entails high communication complexity. Also, Yi et al. (2013) model reveals the credentials when both the servers compromised. As a nutshell, all existing two-server protocols disclose the information when both the servers are compromised by the intruder. Further, device requirement is a major concern in some of the protocols.

Kumari, Sadasivam, and Akash (2016) proposed a 3D ECC PAKE protocol by employing the virtues of plane geometry with ECC encryption technique to offer strong security against server spoofing attacks. Proposed protocol provides equivalent security analogous to Kumari et al. protocol where the strength is based upon the Decisional Diffie-Hellman (DDH) discrete logarithm technique and is proven to be secure. 3D PAKE protocol has been tested for a healthcare application (Kumari, Sadasivam, & Rohini, 2016) and can be applied to similar E-medical applications (Rajan, 2015). Table 1 summarizes the merits and demerits of conventional two-server PAKE protocols.

Table. 1. Comparative analysis of two-server PAKE protocols.

Two-server PAKE protocol	Merits	Limitations
A practical password based two-server authentication and key exchange (Yang et al., 2006)	-Secure against active outside adversary attacks	-Secure channel is required for communication -Back-end server is not robust against impersonation attacks by the active adversary -Back-end server computes the session key established between client and front-end server -Password is revealed when both the servers are compromised
Secure and efficient password-based authenticated key exchange protocol for two-server architecture (Lee & Lee, 2007)	-Secure against server spoofing attacks and stolen verification attacks -Front-end servers do not store any information related to the user's password in the database	-Computational cost is high -Password is revealed when both the servers are compromised
An efficient password-only two-server authenticated key exchange system (Jin et al., 2007)	-Secure against offline dictionary attacks -Session key computation is not possible by back-end server	-Computational complexity is slightly high -Equal contribution is not provided by front-end and back-end servers -Password is revealed when both the servers are compromised
An efficient password based two-server authentication and pre-shared key exchange system using smart cards (Chouksey & Pandey, 2013)	-Secure against offline dictionary attacks, replay attacks, malicious server attacks and man-in-the-middle attacks	-Impersonation of the card reader is possible -Password is revealed when both the servers are compromised
Dynamic identity based authentication protocol for two-server architecture (Sood, 2012)	-Secure against the malicious server attacks, malicious user attacks, stolen smart card attacks, replay attacks and offline dictionary attacks	-Server recognizes expired nonce -Password is revealed when both the servers are compromised
Two-server password-only authenticated key exchange (Katz et al., 2005)	-Rigorous proof of security -Secure against offline dictionary attacks -Symmetric protocol	-Computational and communication complexity is very high -Password is revealed when both the servers are compromised
Efficient two-server password-only authenticated key exchange (Yi et al., 2013)	-Secure against offline dictionary attacks -Symmetric protocol	-Requirement of gateway -Password is revealed when both the servers are compromised

3. PROPOSED METHODOLOGY

3D PAKE protocol is coined based on tetrahedron properties and Diffie-Hellman key exchange mechanism. Existing two-server PAKE protocols assume that both the servers must not compromise together to protect the credentials against invasive attacks. The thought provoking process behind the 3D PAKE is to break the assumption and to offend offline dictionary attacks and assumption and to offend offline dictionary attacks and impersonation attacks caused by an inside/outside adversary. Yang et al. (2006) is modified in the proposed 3D PAKE, to avoid the impersonation of back-end server S2 as front-end server S1 in obtaining the key and the password. The advantages of the proposed methodology are illustrated by considering communication complexity, computational complexity as the metrics. Diffie-Hellman key exchange algorithm process is explained below:

```
Algorithm: Diffie-Hellman Key Exchange
Step 1: Choose an integer group $Z_p^*$ under multiplication modulo 'p' such that 'p' is sufficiently a large prime number.
Step 2: Choose a generator/base point 'g', such that 'g' is a quadratic residue of $Z_p^*$ by satisfying the condition $1 \leq g < p-1$. Generator selection algorithm is as follows:
Algorithm: Generator Selection
For each $g \in Z_p^*$,
Check whether 'g' is a QR of $Z_p^*$
If satisfied, $\forall x \in Z_p^*$, $\exists$ 'i' such that $x = g^i \bmod p$ where $x < p-1$ and $i \geq 0$. Else, 'g' is a QNR of $Z_p^*$.
Step 3: $User_1$ randomly chooses an integer 'a' in $Z_p^*$ and computes $x = g^a$, while $user_2$ chooses an integer 'b' in $Z_p^*$ and computes $y = g^b$, where a, b are considered as private keys and x, y as public keys.
Step 4: $User_1$ and $user_2$ mutually exchange 'x' and 'y'.
Step 5: $User_1$ and $user_2$ compute the secret key as $K_1 = y^a = g^{ba}$ and $K_2 = x^b = g^{ab}$, where $K_1 = K_2$.
```

DH relies on the assumption that no efficient algorithm exists to ascertain the values of 'a', 'b' from gab, if 'a', 'b' and 'g' are chosen randomly and independently (Boneh, 1998). Minimum length of prime number recommended for DH key exchange is 1024-bits to prevent the incidence of any harmful attacks. DH algorithm is secure against passive adversary's attacks. It is not possible by a passive adversary to obtain the secret key based on the observation of data exchanged between user1 and user2. On the other hand, the active attack is possible in DH key exchange as it is a non-authenticated key exchange protocol. To avoid active attacks, the DH key agreement must be put into practice along with strong authentication mechanisms. PAKE protocol is found to be secure against man-in-the-middle attack using low entropy passwords. Thus, the proposed research work is framed with the aid of a PAKE protocol with DH mechanism. Incorporating trigonometric properties further enhance the security of the DH PAKE protocol in fighting against the incidence of all possible active attacks.

3.1 ARCHITECTURE

The 3D PAKE protocol is unconditionally secure, as the password cannot be obtained when both the servers compromise together. Entities used in the 3D PAKE protocol are client C, server S1 and server S2. The protocol executes in three phases, namely, initialization, registration, authentication and key exchange. The notations used in the 3D PAKE protocol are:

Z_p^* – Integer Group 'G' under multiplication modulo 'p'; p – A large prime number
QR_p – Set of quadratic residues modulo 'p'
g_1, g_2, g_3, g_4 – Generators of group Z_p^* of satisfying the QR condition $(b)^2 \equiv g_i \bmod q$ $_{i = 1, 2, 3, 4}$, where $b \in Z_p^*$
x_1, x_2 – Private keys $\in Z_p^*$
y_1, y_2 – Public keys
$b_1, b_2, b_3, a_1, a_2, r, r_1, r_2 \in Z_p^*$
P – Password
$Hash()$ – Secure one-way hash function
$b_4 = b_3 \oplus Hash(P)$
K/K' – Secret key
θ – Angle between the medians of tetrahedron
ω – Circumcenter of the tetrahedron
γ – Adversary
∀ – for all
∃ – there exists

3.1.1 Initialization phase of 3D PAKE protocol

In the initialization phase, the public parameters $\{Z_p{}^*, p, g_1, g_2, g_{3,}Hash\}$ are accepted and disseminated collaboratively by the entities client C, server S1 and S2.

Security of the protocol is based on the generators, prime order and the hash function. The impressive ability is the randomness of the hash function and the generator's discrete logarithm problem. g_4 is a value known only to S1 to avoid man-in-the-middle and client impersonation attacks.

3.1.2 Registration phase of 3D PAKE protocol

The client C selects a password P and compute g_2^P. Further, the client computes b_4 *as* $b_4 = b_3 \oplus Hash(P)$ and forwards the authentication information $\{Username, g_2{}^P, b_3, b_4\}$ to server S1. The server S1 build a tetrahedron from $g_4{}^{g_2{}^P}$ by splitting the value $g_4{}^{g_2^P}$ into $x_1, x_2, x_3, y_1, y_2\ , y_3, z_1, z_2, z_3$ where g_4 is a value known to S1 to avoid impersonation attack. S1 calculate the angle between the medians (ϑ) and circumcenter (ω). Further, it stores ϑ as g_2^θ along with b_3 and transmit username, g_2^ω, b_4 to the server S2. S2 receives and store g_2^ω along withb_4. As a result, registration of client with server S1 and S2 is successful. The operations involved in the registration phase are clearly illustrated in Figure 1.

3.1.3 Authentication Phase of 3D PAKE Protocol

The user induces the verification by sending the username and g_2^P to the server S1, where 'P' is clients' password. Server S1 constructs the tetrahedron from $g_4{}^{g_2^P}$ and ascertains angle between the medians (θ) and circumcenter (ω). The calculated angle between the medians g_2^θ is verified with the storedg_2^θ. Further, S1 forwards the request message $\{Username,\ g_2^\omega\}$ to the server S2.

Upon receiving the message, the server S2 verifies the received g_2^ω against the storedg_2^ω. If the verification is successful, S2 forwards the g_2^ω value to S1 for verifying the authenticity of S2. On the other hand, the server S1 computes a secret key and passes the parameter 'H' to the client. With the received key generation parameter, the client validates the server. Finally, the client and server S1 generate a secret key as shown in Figure 2.

4. SECURITY ANALYSIS

In most cases, the success of a cryptographic attack is based on finding weaknesses in the structure of the protocol. Based on the model and security definition, a particular scheme can be analyzed against attacks to be provable from the state of definition. Proof of correctness, proof of resistance of the protocol against passive attacks, active attacks, offline dictionary attacks and security compliance are discussed in this section.

4.1 PROOF OF CORRECTNESS OF 3D PAKE PROTOCOL

Statement: 3D PAKE protocol is correct if $K = K'$.

Proof:
In server side, S1 computes key K from A, S_2 and S_s,
where $A = g_1{}^a$, $S_2 = A^{b_2} = g_1{}^{ab_2}$ and $S_s = A^{b_1}S_2 = g_1^{ab_1}g_1^{ab_2}$
$= g_1^{a(b_1+b_2)}$
$K = Hash\ (S_s, 1)$

In client side key K' is computed from B and S'_u,
where $B = g_1^{(b_1+\,b_2)}$
Therefore, $S'_u = (B)^a$
$= \left(g_1^{(b_1+\,b_2)}\right)^a$
$= g_1^{a(b_1+b_2)}$
Key $K' \ = Hash\ (S'_u, 1)$
As $K = K'$, the protocol is proven for its correctness.

The random oracle model (Bellare & Rogaway, 1993) is used by the research community to evaluate the security schemes that are constructed using hash functions. In the random oracle model, the behaviour of a hash function is imitated by a deterministic and a proficient function that yields consistently distributed arbitrary values. The 3D PAKE protocol is secure under random oracle model, as the hash value generated is random and irreversible.

4.2 3D PAKE PROTOCOL RESISTANCE TO PASSIVE ATTACKS

Theorem 1: Under the random oracle model, the proposed 3D PAKE protocol is defensive against passive attack with a collision-resistant hash function '*Hash*'.

Proof:

Consider that an adversary γ monitors all the communications between S1 and C and between S1 and S2. Let's contradictorily prove this, by taking into consideration that the messages exchanged between S1, S2 and client C are traced by γ. Even though γ is able to read the messages of S1 and C; S1 and S2, obtaining the password from g_2^P is infeasible, as it is a discrete logarithm problem and there exists no efficient algorithm for quantum computers to obtain a solution for discrete logarithm problem. In a similar sense, if γ obtains g_2^θ / g_2^ω (i.e.,) V_1/V_2 from the messages M2/M3, it is impossible for the adversary to obtain θ/ω. In addition, obtaining a, b_1, b_2 from *A, B2, B, S1, S2* is quite challenging. It is impossible to obtain the vertices of the triangle from the circumcenter (ω) and the angle between the medians (θ). A random one-way hash function is used for transmitting messages between the peers. Hence, '*Hash*', b_4 and S_u is said to be secure under the random oracle model. Thus, a passive attacker γ unable to obtain the password P and the secret key$K$. Hence the proposed protocol is proven to be defensive against passive attack.

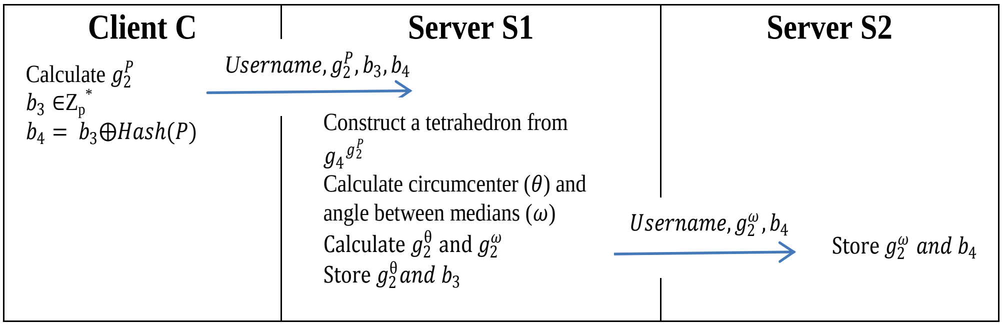
Client C
Server S1
Server S2
Calculate g_2^P
$b_3 \in Z_p^*$
$b_4 = b_3 \oplus Hash(P)$
$Username, g_2^P, b_3, b_4$
Construct a tetrahedron from
$g_4^{g_2^P}$
Calculate circumcenter (θ) and
angle between medians (ω)
Calculate g_2^θ and g_2^ω
Store g_2^θ and b_3
$Username, g_2^\omega, b_4$
Store g_2^ω and b_4

Fig. 1. A detailed registration process of 3D PAKE protocol.

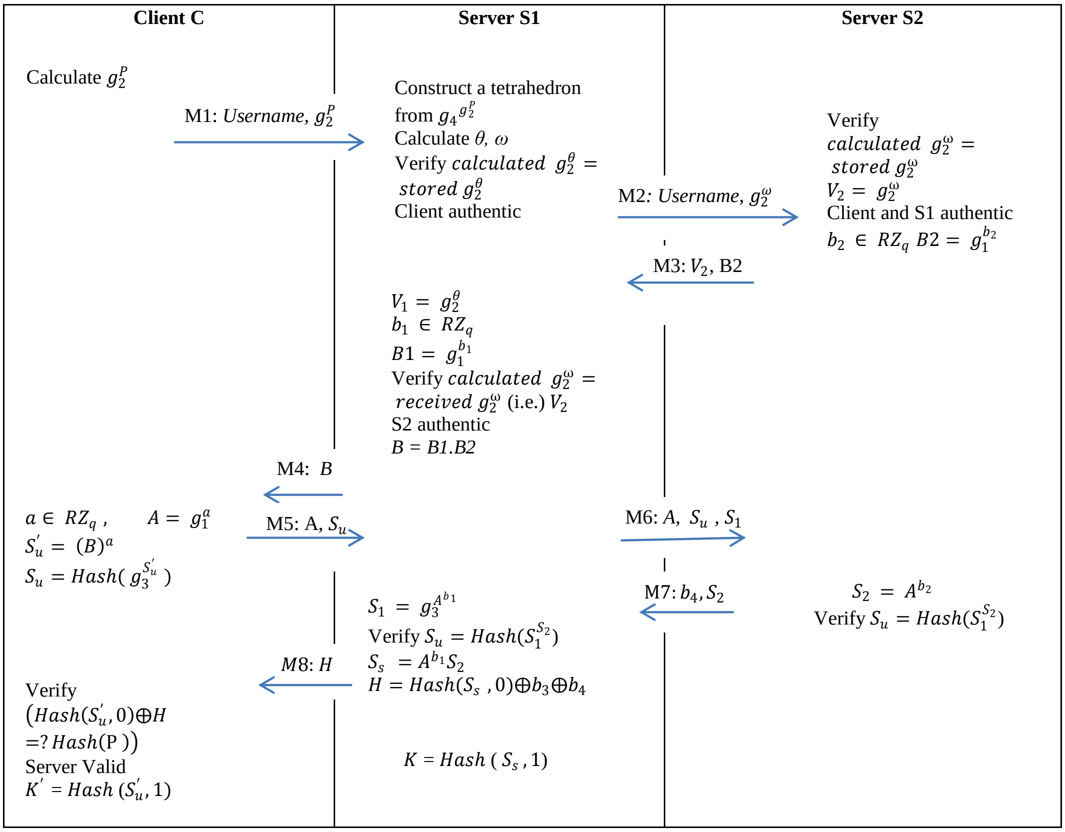

Fig. 2. A detailed authentication and key exchange process of 3D PAKE protocol.

4.3 3D PAKE PROTOCOL RESISTANCE TO ACTIVE ATTACKS

Theorem 2: The proposed 3D PAKE protocol is defensive against active attack, if there is no existence of polynomial-time algorithm to break the Discrete Logarithm Problem (DLP).

Proof:

Assumption (i): Assume that an active adversary γ impersonate as client C by compromising server S1/S2.

a. Assume that an active adversary γ modifies g_2^P as $g_2^{P'}$. Let's contradictorily prove this, by taking into consideration that an active adversary γ has compromised server S1/S2 to impersonate as client C, by replacing/modifying g_2^P transferred in message M1 with arbitrary number $g_2^{P'}$instead of g_2^P. Since the challenger receives $g_2^{P'}$instead of g_2^P, client verification fails at server side as per Equation (1).

Verify $calculated\ g_2^{\theta} = \ stored\ g_2^{\theta}$ in S1

Verify $calculated\ g_2^{\omega} = \ stored\ g_2^{\omega}$ in S2 (1)

where 'θ' and 'ω' are derived from $g_4{}^{g_2{}^P}$.

b. Assume that adversary γ modifies the value S_u transferred in message M5 as S_{uv}. Since, the challenger receives S_{uv} instead of S_u, establishment of key is liable to failure in server S1 side as per Equation (2).

$$S_u = Hash(S_1^{S_2}) \tag{2}$$

c. Further, imagine that the adversary γ is assuming $A \rightarrow (g_1^a)$ as $A' \rightarrow (g_1^{a'})$ transferred in message M5 for the key generation in server S1. Since, the challenger receives A' instead of A, verification of server is liable to failure on client side as per Equation (3).

$$Hash(S_u', 0) \oplus H = ?\, Hash(\mathrm{P}\ ) \tag{3}$$

In server side, $K = Hash(S_s) = Hash(A'^{b_1}S_2\ ) = Hash\left(g_1^{a'b_1}S_2\ \right) = Hash\left(g_1^{a'b_1b_2}\right)$

In client side, $K' = \ Hash\ \ (S_u') \ = \ Hash\,(B^a) \ = \ Hash\ \ (g_1^{a\,b_1b_2})$

Therefore, $K \neq \ K'$.

Analysis:

Considering the case $A \rightarrow (g_1^a)$ as $A' \rightarrow \ (g_1^{a'})$, S_u as S_{uv} and g_2^P as $g_2^{P'}$, the active adversary γ cannot succeed in generating the secret key K, such that K= K'.

Assumption (ii): Assume that an active adversary γ impersonate as server S1 by compromising server S2.

a. Assume that an active adversary γ modifies g_2^P as $g_2^{P'}$. Let's contradictorily prove this, by taking into consideration, that an active adversary γ has compromised the server S2 to impersonate as server S1 by replacing/modifying the messages exchanged between the server and the client. Such an adversary may modify the value g_2^P transferred in message M1 with a random number. Authentication and key exchange process terminates as proved in Assumption (i): case (a) of Theorem 2. Challenger tries to construct the triangle from $g_4{}^{g_2^P}$ and examines whether $calculated\ g_2^{\theta} = stored\ g_2^{\theta}$. As an effect, triangle construction is not possible by γ, as the value g_4 is not known to the adversary.

b. The adversary γ tries to modify the values transferred in messages M4: *B*, M5: *A*, S_u and M8: *H*. Challenger verifies whether $S_u = Hash(S_1^{S_2})$ and computes S_s and H as, $S_s = A^{b_1}S_2$ and $H = Hash(S_s, 0) \oplus b_3 \oplus b_4$. Retrieving the value b_3 is impossible by γ, as the value is stored in server S1. Modifications in messages M4, M5 or in M8, leads to termination of the key generation process as per Assumption (i): case (a) and (b) of Theorem 2.

Analysis:

Thus, by modifying the values in messages M4: *B*, M5: *A*,S_u, M8: *H* and g_2^P as $g_2^{P'}$, the active adversary γ can't prevail in generating the secret key *K*.

Assumption (iii): Assume that an active adversary γ impersonate as server S2 by compromising server S1.

a. Assume that an active adversary γ has compromised the server S1 to impersonate as server S2 by replacing/modifying the messages exchanged between the server and the client. Such an adversary may modify the value g_2^{ω} transferred in message M2 with a random number. Challenger verifies

$received\ g_2^{\omega}$ with $stored\ g_2^{\omega}$. As an effect, retrieving the stored g_2^{ω} value is impossible by γ, since, the value is known only to server S2.

b. The adversary **γ** may try to modify the values transferred in messages M6: $A,\ S_{u,},\ S_1$ or M7: $b_{4,}$, S_2. Challenger computes $S_2 = A^{b_2}$ and verifies whether $S_u = Hash(S_1^{S_2})$. Retrieving value b_4 is impossible by γ, since, the value is stored in server S2. Altering the values in messages M6/M7, terminates the key generation process as proved in Assumption (i): case (a) and (b) of Theorem 2.

Analysis:

Thus, by modifying the values of the messages M6: $A,\ S_{u,}, S_1$, M7: $b_{4,}, S_2$ or g_2^{ω} with a random number by the active adversary **γ** cannot succeed in generating the secret key K.

Remark 1:

Active impersonation of one server as another is possible in Yang et al. (2006) model. 3D PAKE protocol routs the drawback of Yang et al. protocol and proved it is secure against impersonation attacks on server S1 and S2 as shown by Theorems 1 and 2. When both the servers are compromised by the intruder, it is infeasible to determine the password ‘P’ from the stored values, based on the properties of the tetrahedron. It is demonstrated that the proposed 3D protocol is strong and intractable, when compared to existing two-server PAKE protocols in the circumstance of the servers’ database are controlled by the adversaries.

4.4 3D PAKE PROTOCOL RESISTANCE TO OFFLINE DICTIONARY ATTACKS

Theorem 3: The proposed 3D PAKE protocol is defensive against offline dictionary attack by providing two levels of security.

Proof:

Assumption (i): Assume that an active adversary γ breaks the 3D PAKE protocol under offline dictionary attack.

a. Assurance of primary level of security by β. Let’s contradictorily prove this, by taking into consideration, when the adversary γ attains access to the database of both the servers by dictionary attack, the adversary obtain g_2^{θ} and g_2^{ω} values. However, deriving θ and ω from g_2^{θ} and g_2^{ω} respectively is NP hard. Hence, it cannot be resolved in polynomial time. Thus, primary level of security is guaranteed.

a. Assurance of the second level of security by β. If the adversary γ manages to solve DLP, then θ and ω values are attained by the adversary. However, finding the vertices of the triangle from θ and ω values is not possible, where θ and ω are derived from ${g_4}^{g_2^P}$. Henceforth, second level of security is assured.

The protocol has been tested with Sqlmap, Wireshark, Havij, Vega, Websecurify, Webcruiser, SSLSmart, WSAttacker and WSDigger to affirm the strength of the protocol. In addition, 3D PAKE complies with known key security, forward secrecy, key control, key confirmation, zero-knowledge proof, explicit key authentication, key freshness, impersonation resilience and reciprocity principles. Also, it is sturdy against low-encryption-exponent attack, known and chosen cipher text attack, known and chosen plaintext attack, sniffer attack, replay attack, man in the middle attack and rainbow table attack. Table 2 summarizes the security standards of the proposed protocol and it proves that the proposed protocol is rigid.

5. PERFORMANCE ANALYSIS

The data set used to test the protocol comprises of 100000 passwords. Table 3 shows the experimental results of 3D PAKE Protocol tested for a healthcare application. Password transformation relies upon tetrahedron parameters ω and θ. The value of ω and ϑ shows the prominence of heuristic information and their impacts. Key length adopted in 3D PAKE is 3072-bits for proper regulation and to prevent illegitimate access.

5.1. COMMUNICATION AND COMPUTATIONAL COMPLEXITY OF 3D PAKE PROTOCOL

The performance of the proposed 3D PAKE protocol is analyzed by comparison with the existing two-server PAKE protocols. Number of group elements in communication are measured in terms of ‘L’ and the number of hash values in communication is measured in terms of 'l'. The communication complexity includes number of group elements in communication, the number of hash values in communication and the number of rounds taken by the protocol for successful completion.

Communication complexity of 3D PAKE is 9L + 4l and computational complexity is 32, which is very near to that of existing protocols as presented in Table 4. Slight increase in computation is due to the construction of the tetrahedron. It is noticed that the client side complexity is considerably reduced. Furthermore, as the proposed protocol is asymmetric, there is a notable difference in the server side because of the communication between the servers S1 and S2. However, this computational complexity can be negotiated as the server S2 is hidden and protected from security vulnerabilities. Nevertheless, it routs the postulation made by other protocols and augments the security.

Table. 2. Functionality comparison of 3D PAKE protocol with Yang et al. and Yi et al. protocol.

Functionality	(Yang et al., 2006) **Protocol**	(Yi et al., 2013) **Protocol**	**3D PAKE protocol**
Known key security	Yes	Yes	Yes
Forward secrecy	Yes	Yes	Yes
Key control	Yes	Yes	Yes
Key confirmation	Yes	Yes	Yes
Zero-knowledge proof	Yes	Yes	Yes
Explicit key authentication	Yes	Yes	Yes
Key freshness	Yes	Yes	Yes
Reciprocity	Yes	Yes	Yes
Impersonation resilience	Yes	No	No
Low-encryption-exponent attack	Possible	Possible	Possible
Known and chosen ciphertext attack	Possible	Not Possible	Not Possible
Known and chosen plaintext attack	Possible	Not Possible	Not Possible
Sniffer attack	Possible	Not Possible	Not Possible
Replay attack	-	-	Restricted
Man in the middle attack	Not Possible	Not Possible	Not Possible
Impersonation attack by inside adversary	Possible	Not Possible	Not Possible
Offline dictionary attacks on servers database to disclose the password	Possible	Possible	Not Possible
Online dictionary attack	-	-	Restricted
Known-key distinguishing attack	Not Possible	Not Possible	Not Possible
Chosen-key distinguishing attack	Not Possible	Not Possible	Not Possible
Interleaving attack	Not Possible	Not Possible	Not Possible
Lowe's attack	Not Possible	Not Possible	Not Possible
Cross-site scripting attack	-	-	Restricted
SQL injection attack	-	-	Restricted
Side channel attack	-	-	Restricted but not limited
Rainbow table attack	-	-	Restricted

Table. 3. Test cases of the proposed 3D PAKE protocol.

S.No	Username	Password	Theta (θ)	Omega(ω) [x,y,z]	Run Time (ns)	Session Key (bits)
1	Mary23	yaguacire95	0.541823456	[7.4550984849506285, -0.0339898996708528, -3.668694444818968]	5.01567823E8	554c5e325b2ca99c5e8e5549bcb5ac1bbad0671c3dd5ed84dace0b47aa00f91b0d162c6c30eb594c5de404e6a5d1cdb884fec30fdcbd3c7a36da60f45f7ef58d
2	David	re1ns+@ll	0.353009486	[3.0106024072279522, 7.9681927783161415, 3.6914056168652216]	5.21056267E8	3222145f3D8aa569f47f9d8d087a3f70ffff965607b2cf14581936d1b348f0622bff807946884d2432fcbb33a21a9bee75f4c2add8147f554708b90e80cd608a
3	Dev	tenant-atwill	0.416137519	[-934.4125473274593, -572.3536119949651, 50.00000000000001]	4.53459167E8	68c7f40ef122548eb6188505288058cc4957cf89027a1f9bf3deb1eabe9c81fa860dbb09c6ef59404d96d576d66070c326a63b4cddf471a014019f804bff2f21
4	antony3	rebecca	1.552935703	[15.672995055568377, 0.0817512361079066, 52.295621909730245]	4.56201638E8	b030bc87675e46b4084ed62a4e1d188d1fde30bf8a5d9e7ae2e3f9c8f15cca016d21dc4b0779f79531c93D2c1b7d9a709cdf8c357e6d58e0a0da3571a921a767
5	joshua	un!ver$a1!+y	0.584438919	[-5163.067772650183, -1989.9957515875855, -266.4610155800105]	4.35381735E8	02ea6eab7a895fea9d407066ff6f9bb7a226f7a9fcd598085cb987cf1f7e9098d317eb11a1118ecf4c60c9bd4306a06b15ea4ff907acd70945247231ef9b6bc3

For a clear understanding, values are graphically presented in Figure 3. From Figure 3, it can be inferred that 3D PAKE provides a fair communication complexity. For a broad computational cost analysis, the number of transmissions, hash computations, modular/scalar multiplications, XOR operations and modular exponentiations are examined. The proposed 3D PAKE protocol computation wise performs in a fair manner when compared to Yang et al. (2006), Yi et al. (2013), and Jin et al. (2007) protocols as shown in Table 5.

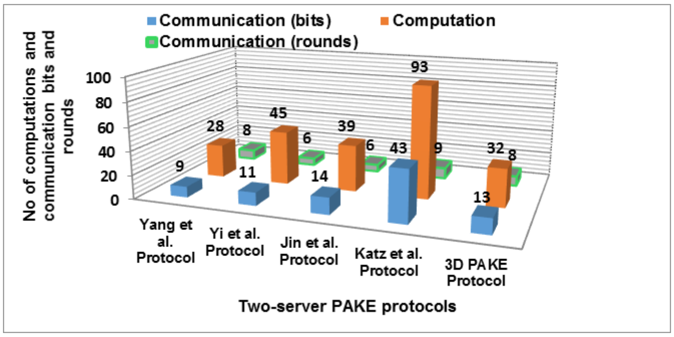

Fig. 3. Complexity analysis of 3D PAKE protocol.

Table. 4. Communication and computational complexity analysis of 3D PAKE protocol.

Participants	**Yang et al. (2006) Protocol** [***ASYMMETRIC***]	**Yi et al., (2013) Protocol** [***SYMMETRIC***]	**Jin et al. (2007) Protocol** [***ASYMMETRIC***]	**Katz et al. (2005) Protocol** [***SYMMETRIC***]	**3D PAKE Protocol** [***ASYMMETRIC***]
Client: Communication (bits)	2L + 2l	3L + 4l	6L + 2l	15 L	3L + 2l
Client: Communication (rounds)	4	4	3	3	4
Client: Computation	7	21	12	34	9
Server S1: Communication (bits)	6L + 3l	6L + 3l	11L + 3l	14L	9L + 4l
Server S1: Communication (rounds)	8	5	6	3	8
Server S1: Computation	15	12	19	27	19
Server S2: Communication (bits)	4L + 1l	6L + 3l	5L + 1l	14L	6L + 2l
Server S2: Communication (rounds)	4	5	3	3	4
Server S2: Computation	6	12	8	27	4
	Comm: **9** **(6L+3l)** Client – S1 – S2	Comm: **11** **(7L+4l)** Client – S1 Client – S2	Comm: **14** **(11L + 3l)** Client – S1 – S2	Comm: **43** Client – G-S1 Client – G-S2	Comm: **13** **(9L + 4l)** Client – S1 – S2
	Comp:**28** Client – S1 – S2	Comp:**Worst case: 45** **Best case: 33** Client – S1 Client – S2	Comp:**39** Client – S1 – S2	Comp:**Worst case: 93** **Best case:66** Client – G-S1 Client – G-S2	Comp: **32** Client – S1 – S2
	Rounds: **8** Client – S1 – S2	Rounds: **6** Client – S1 Client – S2	Rounds: **6** Client – S1 – S2	Rounds: **9** Client – G-S1 Client – G-S2	Rounds: **8** Client – S1 – S2

Table. 5. Comparative cost analysis of 3D PAKE protocol.

Cost Computation Parameters	Yang et al. (2006) Protocol	Yi et al. (2013) Protocol	Jin et al. (2007) Protocol	3D PAKE Protocol
No. of transmissions	8	6	6	8
No. of hash computations	7	15	8	8
No. of modular/scalar multiplications	5	3	7	6
No. of modular exponentiations	16	16	24	15
No. of XOR operations	0	11	0	3
No. of authentication parameters	1	1	1	2

Thus, the proposed 3D PAKE performs judiciously computation wise. To the best of our cognizance, a foolproof two-server 3D PAKE protocol is proposed based on tetrahedron properties and proved its resistance against attacks.

6. CONCLUSION

A formal design and evaluation of a state-of-art tetrahedron (3D) based two-server PAKE protocol is presented in this paper with definite proof of security. With the assistance of ω and ϑ parameters, offline dictionary attacks occurring on the server's database are proclaimed as a challenge as rightly pointed and proved in section 4.4; thereby, obtaining the password is infeasible when both the servers are compromised. This assures the robustness of the protocol against dictionary attack in 3D. It is also observed, that the 3D PAKE protocol is performing reasonably well in communication and computation, as discussed in section 5.1. As a future avenue of research, the proposed 3D PAKE protocol security can be reinforced constantly by adding additional parameters / shapes with formal proof of security.

CONFLICT OF INTEREST

The autors have no conflicts of interest to declare.

REFERENCES

Abdalla, M., Chevassut, O., Fouque, P. A., & Pointcheval, D. (2005). A simple threshold authenticated key exchange from short secrets. *Lecture Notes in Computer Science. 3788*, 566-584.

Bellare, M., & Rogaway, P. (1993). Random oracles are practical: A paradigm for designing efficient protocols. *1st ACM Conference on Computer and Communications Security*, pp. 62-73.

Bellovin, S. M., & Merritt, M. (1990). Limitations of the Kerberos authentication system. ACM SIGCOMM Computer Communication Review, 20(5), 119-132.

Bellovin, S. M., & Merritt, M. (1992). Encrypted key exchange: Password-based protocols secure against dictionary attacks. *IEEE Proceedings of the Symposium on Security and Privacy*, (pp. 72-84). IEEE.

Boneh, D. (1998). The decision diffie-hellman problem. *Lecture Notes in Computer Science, 1423*, pp. 48-63.

Byun, J. W., Lee, D. H., & Lim, J. I. (2006). Security analysis and improvement of a gateway-oriented password-based authenticated key exchange protocol. *IEEE Communications Letters*, *10*(9), 683-685.

Chien, H. Y., Wu, T. C., & Yeh, M. K. (2013). Provably secure gateway-oriented password-based authenticated key exchange protocol resistant to password guessing attacks. *Journal of Information Science and Engineering*, *29*(2), 249-265.

Choate J. (1976). Tetrahedral Treats, Available from: http://www.zebragraph.com/Geometers_Corner_files/tetrahedral treats.pdf

Chouksey, A., & Yogadhar, P. (2013). An efficient password based two-server authentication and pre-shared key exchange system using smart cards. *International Journal of Computer Science and Information Technologies*, *4*(1), 117-120.

Dennis F. (2012). Final Report on Diginotar Hack Shows Total Compromise of CA Servers, Available from: https://threatpost.com/final-report-diginotar-hack-shows-total-compromise-ca-servers-103112/77170/

Jack, D. (2008). Construction of a Triangle from Circumcenter. (2008) Orthocenter and Incenter, Available from: https://www.cut-the-knot.org/triangle/O-H-I.shtml

Jin, H., Wong, D. S., & Xu, Y. (2007). An efficient password-only two-server authenticated key exchange system. *Lecture Notes in Computer Science,* (pp. 44-56).

Katz, J., MacKenzie, P., Taban, G., Gligor, V. (2005) Two-server password-only authenticated key exchange. *Lecture Notes in Computer Science*, 3531, pp. 1-16.

Kumari, K. A., Sadasivam, G. S., & Akash, S. A. (2016). A Secure Android Application with Integration of Wearables for Healthcare Monitoring System Using 3D ECCDH PAKE Protocol. *Journal of Medical Imaging and Health Informatics*, *6*(6), 1548-1551.

Kumari, K. A., Sadasivam, G. S., & Rohini, L. (2016). An Efficient 3D Elliptic Curve Diffie–Hellman (ECDH) Based Two-Server Password-Only Authenticated Key Exchange Protocol with Provable Security. *IETE Journal of Research*, *62*(6), 762-773.

Lee, J. H., & Lee, D. H. (2007). Secure and efficient password-based authenticated key exchange protocol for two-server architecture. *International Conference on Convergence Information Technology,* (pp. 2102-2107). IEEE.

Lin, C. L., Sun, H. M., & Hwang, T. (2000). Three-party encrypted key exchange: attacks and a solution. *ACM SIGOPS Operating Systems Review, 34*(4), 12-20.

MacKenzie, P., Shrimpton, T., & Jakobsson, M. (2002). Threshold password-authenticated key exchange. *Lecture Notes in Computer Science*, *2442*, 385-400.

Pooja, D., Shilpi, G., Sujata, S., & Vinita, G. (2012). Secured authentication: 3d password. *International Journal of Engineering and Management Sciences*, *3*(2), 242-245.

Rajan, S. (2015). Review and investigations on future research directions of mobile based telecare system for cardiac surveillance. *Journal of applied research and technology*, *13*(4), 454-460.

Sood, S. K. (2012). Dynamic identity based authentication protocol for two-server architecture. *Journal of Information Security*, *3*, 326.

Wan, Z., Deng, R. H., Bao, F., & Preneel, B. (2007). nPAKE+: A hierarchical group password-authenticated key exchange protocol using different passwords. *Lecture Notes in Computer Science, 4861,* 31-43.

Yang, Y., Deng, R. H., & Bao, F. (2006). A practical password-based two-server authentication and key exchange system. *IEEE Transactions on Dependable and Secure Computing*, *3*(2), 105-114.

Yi, X., Ling, S., & Wang, H. (2013). Efficient two-server password-only authenticated key exchange. *IEEE transactions on Parallel and Distributed systems*, *24*(9), 1773-1782.