Hash Based Quantum Key Web User Password Security in Two Server

[1]**T.S.Thangavel and** [2]**Dr. A. Krishnan**

[1]AP/ Dept of M.C.A, K.S.Rangasamy College of Technology,
[2]Dean, K.S.Rangasamy College of Technology, Tamilnadu, India.

Summary
The authentication systems which uses passwords to authenticate their systems stores their password in a central server which is easily prone to attack and if they are being compromised by the intruder, it is possible for the intruder to obtain the password and gain access to the contents of the user. To overcome this problem, the multi-server systems were being proposed in which the user has to communicate in parallel with several or all of the servers for the purpose of authentication. Such system requires a large communication bandwidth and needs for synchronization at the user. The system is not easy to deploy and maintain or it requires the protocols which are quite expensive. To overcome these problems the two server authentication system proposed here uses only the passwords and the session keys rather than performing any cryptographic techniques. The two server system is particularly suitable for resource-constrained users due to its efficiency in terms of both computation and communication. With the itricate security principle of quantum theory and traditional public key model, integration is made to provide an improved security model for password authentication between the password exchange of two servers.The proposed work presented a user friendly secured password authentication system with two servers by applying quantum cryptographic. To start with, built user friendly browser extension password hash transparently produces a different password for each site, improving web password security. To improve the single server security issues, construct an efficient two server password authentication in terms of computation and communication. Finally quantum key cryptographic techniques are integrated to hash mechanism in two server authenticity to easily resist replay and passive attacks. User authentication and session key verification can be accomplished in one step without public discussions between a sender and receiver. The performance of integrated Quantum Key Distribution (QKD) systems and classical public key model have shown experimentally better performance in terms of computational efficiency and security rounds than traditional cryptic security model.

Key words:
Hash function, Pseudo Random number, Service Server, Control server, Two Server Password Authentication, Quantum Key Distribution, Session Key

1. Introduction

The flaws in chaotic systems make the potential harmful choices for use in random number generators for cryptographic security systems in generating passwords. Transforming the state of chaotic system into a random number is a much slower process than typical computation. Repeatedly generating random numbers from such a system can become a time bottleneck. A pseudo-random number generator deterministically generates a sequence of numbers by some computational process from an initial number, called a seed. The goal of the computational process is to generate a sequence of numbers from the seed that appear to be random. An outside observer cannot predict the next number to be generated from the list of numbers previously generated without effort. With this hash mechanism is applied to tighten the authentic verification of user password. Most hash password-based user authentication systems place total trust on the authentication server where passwords or easily derived password verification data are stored in a central database. These systems could be easily compromised by offline dictionary attacks initiated at the server side. Compromise of the authentication server by either outsiders or insiders subjects all user passwords to exposure and may have serious problems. To overcome these problems in the single server system many of the systems has been proposed such as multi-server systems, public key cryptography and password systems, threshold password authentication systems, two server password authentication systems.

The proposed work continues the line of research on the two-server paradigm in [10], [11], extend the model by imposing different levels of trust upon the two servers, and adopt a very different method at the technical level in the protocol design. As a result, we propose a practical two-server password authentication and key exchange system that is secure against offline dictionary attacks by servers when they are controlled by adversaries. Moreover, the proposed system is particularly suitable for resource constrained users due to its efficiency in terms of both computation and communication. Computing exponential increase in power requires setting the bar always higher to secure password data transmissions in two server authentication. The ideal solution would transmit data in quantum bits, but truly quantum information processing may lie decades away. Therefore, several companies have

Manuscript received July 5, 2010
Manuscript revised July 20, 2010

focused on bringing one aspect of quantum communications to market quantum key distribution (QKD), used to exchange secret keys that protect data during transmission.

The key distributed using quantum cryptography would be almost impossible to steal because QKD systems continually and randomly generate new private keys that both parties share automatically. A compromised key in a QKD system can only decrypt a small amount of encoded information because the private key may be changed every second or even continuously. To build up a secret key from a stream of single photons, each photon is encoded with a bit value of 0 or 1, typically by a photon in some superposition state, such as polarization. These photons are emitted by a conventional laser as pulses of light so dim that most pulses do not emit a photon. This approach ensures that few pulses contain more than one photon. Additional losses occur as photons travel through the fiber-optic line. In the end, only a small fraction of the received pulses actually contain a photon. However, this low yield is not problematic for QKD because only photons that reach the receiver are used. The key is generally encoded in either the polarization or the relative phase of the photon

In key distribution protocols, two users obtain a shared session key via a trusted center (TC). Since three parties (two users and one TC) are involved in session key negotiations, these protocols are called three-party key distribution protocols, as in contrast with two-party protocols where only the sender and receiver are involved in session key negotiations. In quantum cryptography, quantum key distribution protocols (QKDPs) employ quantum mechanisms to distribute session keys and public discussions to check for eavesdroppers and verify the correctness of a session key. However, public discussions require additional communication rounds between a sender and receiver and cost precious qubits. By contrast, classical cryptography provides convenient techniques that enable efficient key verification and user authentication. These QKDP and classical cryptographic model motivates us to propose an integrated password communication between two server authentication system. The proposal work in this paper provides a pattern of integrating the classical key verification with the quantum mechanism employed in distributing the session key and provide efficient password sharing between the two servers to make the password authentication more robust.

2. Literature Review

In the context of cryptographic applications [19], there may be an hostile trespasser or agent, who desires to infiltrate the security of cryptographic security system in order to gain access to sensitive, confidential, or valuable information contained therein. In order to ensure the utmost security, it is essential that the security system implements a method for generating a random number that appears completely random. In this manner, a completely random password or cryptographic key presents no opening or prior knowledge that can be exploited by an hostile agent. [20] A chaotic system is one with a state that changes over time in a largely unpredictable manner. To use the chaotic system [22] to generate a random number, there is some means of converting the state of the system into a sequence of bits (i.e., a binary number). A pseudo-random binary string can be generated from the digital recording of static noise via a digital microphone. However, there are several problems associated with simply using a chaotic system as a source of random numbers.[21] Furthermore, the behavior of chaotic systems can be far from completely random. With this, hash based pseudo random password were generated to provide a highly authenticated system.

Public key techniques are absolutely necessary to make password systems secure against offline dictionary attacks, whereas the involvement of public key cryptosystems under a PKI (e.g., public key encryption and digital signature schemes) is not essential. There are two separate approaches to the development of secure password systems one is a combined use of a password and public key cryptosystem under a PKI, and the other is a password only approach. In these systems, the use of public keys entails the deployment and maintenance of a PKI for public key certification and adds to users the burden of checking key validity. To eliminate this drawback, password-only protocols (password authenticated key exchange or PAKE) have been extensively studied, e.g., [20], [21], [22]. The PAKE protocols do not involve any public key cryptosystem under a PKI and, therefore, are much more attractive for real-world applications. Any use of public key cryptosystem under a PKI in a password authentication system should be avoided since, otherwise, the benefits brought by the use of password would be counteracted to a great extent.

Most of the existing password systems were designed over a single server, where each user shares a password or some password verification data (PVD) with a single authentication server (e.g., [2], [3], [4]). These systems are essentially intended to defeat offline dictionary attacks by outside attackers and assume that the sever is completely trusted in protecting the user password database. Unfortunately, attackers in practice take on a variety of forms, such as hackers, viruses, worms, accidents, mis-configurations, and disgruntled system administrators. As a result, no security measures and precautions can guarantee that a system will never be penetrated. Once an authentication server is compromised, all the user passwords or PVD fall in the hands of the attackers, who are definitely effective in offline dictionary attacks against the user passwords. To eliminate this single point of vulnerability inherent in the single-server systems, password systems based on multiple servers were proposed. The principle is distributing the password database as well as the authentication function to multiple servers so that an attacker is forced to compromise several servers to be successful in offline dictionary attacks.

The system in [6], believed to be the first multiserver password system, splits a password among multiple servers. However, the servers in [6] need to use public keys. An improved version of [6] was proposed in [7], which eliminates the use of public keys by the servers. Further and more rigorous extensions were due to [8], where the former built a t-out-of-n threshold PAKE protocol and provided a formal security proof under the random oracle model [5] and the latter presented two provably secure threshold PAKE protocols under the standard model. While the protocols are theoretically significant, they have low efficiency and high operational overhead. In these multi-server password systems, either the servers are equally exposed to the users and a user has to communicate in parallel with several or all servers for authentication, or a gateway is introduced between the users and the servers.

Recently, Brainard et al. [9] proposed a two-server password system in which one server exposes itself to users and the other is hidden from the public. While this two-server setting is interesting, it is not a password-only system: Both servers need to have public keys to protect the communication channels from users to servers. As we have stressed earlier, this makes it difficult to fully enjoy the benefits of a password system. In addition, the system in [9] only performs unilateral authentication and relies on the Secure Socket Layer (SSL) to establish a session key between a user and the front-end server. Subsequently, Yang et al. [17] extended and tailored this two-server system to the context of federated enterprises, where the back-end server is managed by an enterprise head quarter and each affiliating organization operates a front-end server.

The most common standard protocol for Quantum Key Distribution QKD is called BB84, it uses a stream of single photons to transfer a cryptographic key between two parties, who can use it to encode and decode data transmitted using standard high-speed techniques. Right now, single photons allow real-time data transmissions only at low speed, typically 100 bits/s—a hundred millionth the speed of today’s fastest fiber-optic transmission systems. That explains why most companies have focused on commercializing QKD and not on data encryption. Polarization-based encoding works best for free-space communication systems rather than fiber-optic lines. Data are transmitted faster in free-space systems, but they cannot traverse the longer distances of fiber-optic links.

In classical cryptography, three-party key distribution protocols [10], [11] utilize challenge response mechanisms [12], [13] or timestamps [14], [15] to prevent replay attacks [16]. However, challenge response mechanisms require at least two communication rounds [9] between the TC and participants, and the timestamp approach needs the assumption of clock synchronization which is not practical in distributed systems (due to the unpredictable nature of network delays and potential hostile attacks) [17]. Furthermore, classical cryptography cannot detect the existence of passive attacks [18] such as eavesdropping. On the contrary, a quantum channel eliminates eavesdropping, and, therefore, replay attacks. This fact can then be used to reduce the number of rounds of other protocols based on challenge-response mechanisms to a trusted center (and not only three-party authenticated key distribution protocols). The proposal in this paper integrates QKDP and classical model, in which TC and a participant synchronize their polarization bases according to a pre-shared secret key in the two server password authentication system. During the session key distribution, the pre-shared secret key together with a random string are used to produce another key encryption key to encipher the session key. A recipient will not receive the same polarization q-bits even if an identical session key is retransmitted.

3. Hash based Pseudo Random Password

Random password generators normally output a string of symbols of specified length. These can be individual characters from some character set, syllables designed to form pronounceable passwords, or words from some word list to form a passphrase. The program can be customized to ensure the resulting password complies with the local password policy, say by always producing a mix of letters, numbers and special characters. The strength of a random password can be calculated by computing the information entropy of the random process that produced it. If each symbol in the password is produced independently, the entropy is just given by the formula

$$H = L\log_2 N = L\frac{\log N}{\log 2}$$

Where N is the number of possible symbols and L is the number of symbols in the password. The function $\log_2$ is the base-2 logarithm. H is measured in bits. An eight character password of single case letters and digits would have 41 bits of entropy (8 x 5.17). Thus a password generated using a 32-bit generator has maximum entropy of 32 bits, regardless of the number of characters the password contains.

3.1 Secure Hashing

The proposed methodology of the secure hash password system contains one-way hash functions that can process a message to produce a condensed representation called a message digest. This algorithm enables the determination of a message's integrity, any change to the message will, with a very high probability, results in a different message digest. This property is useful in the generation and verification of digital signatures and message authentication codes, and in the generation of random numbers. The algorithm is described in two stages, preprocessing and hash computation. Preprocessing involves padding a message, parsing the padded message into m-bit blocks, and setting initialization values to be used in the hash computation. The hash computation generates a message schedule from the padded message and uses that schedule, along with functions, constants, and word operations to iteratively generate a series of hash values. The final hash value generated by the hash computation is used to determine the message digest. The design principle of hash functions is iterating a compression function (here denoted F), which takes as input s bits and returns r bits (with s > r). The resulting function is then chained to operate on strings of arbitrary length (Fig 1). The validity of such a design has been established and its security is proven not worse than the security of the compression function. The core of the compression function is a random binary matrix H of size r×n. The parameters for the hash function are n the number of columns of H, r the number of rows of H and the size in bits of the function output, and w the number of columns of H added at each round.

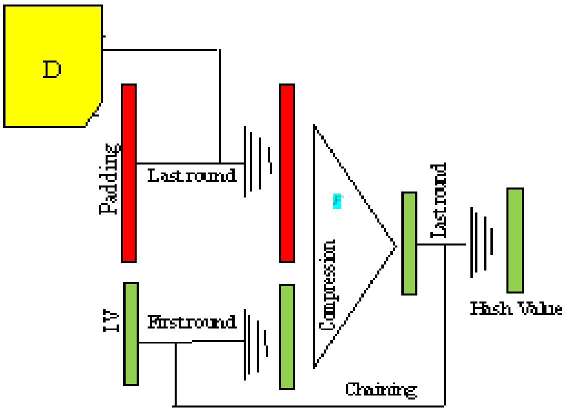


Fig 1: Iterative hash function structure Compression Hash function Algorithm

Input : s bits of data.

1. Split the S input bits in w pars S1 . . . . . Sw of $\log_2\left(\frac{n}{w}\right)$ bits.

2. Convert each S_i to an integer between 1 and $\frac{n}{w}$.

3. Choose the corresponding column in each H_i ;

4. Add the w chosen columns to obtain a binary string of length r.

Output: r bits of hash.

4. Two Server Password Authentication System

Three types of entities are involved in our system, i.e., users, a service server (SS) that is the public server in the two server model, and a control server (CS) that is the back-end server. In this setting, users only communicate with SS and do not necessarily know CS. For the purpose of user authentication, a user U has a password which is transformed into two long secrets, which are held by SS and CS, respectively. Based on their respective shares, SS and CS together validate users during user login. CS is controlled by a passive adversary and SS is controlled by an active adversary in terms of offline dictionary attacks to user passwords, but they do not collude (otherwise, it equates the single-server model).

A passive adversary follows honest-but-curious behavior, that is, it honestly executes the protocol according to the protocol specification and does not modify data. But it eavesdrops on communication channels, collects protocol transcripts and tries to derive user passwords from the transcripts. Moreover, when an passive adversary controls a server, it knows all internal states of knowledge known to the server, including its private key (if any) and the shares of user passwords. In contrast, an active adversary can act arbitrarily in order to uncover user passwords. Besides, we assume a secret communication channel between SS and CS for this basic protocol. This security model exploits the different levels of trust upon the two servers. This holds with respect to outside attackers. As far as inside attackers are concerned, justifications come from our application and generalization of the system to the architecture of a single control server supporting multiple service servers, where the control server affords and deserves enforcing more stringent security measurements against inside attackers. The back-end server is strictly passive and is not allowed to eavesdrop on communication channels, while CS in our setting is allowed for eavesdropping.

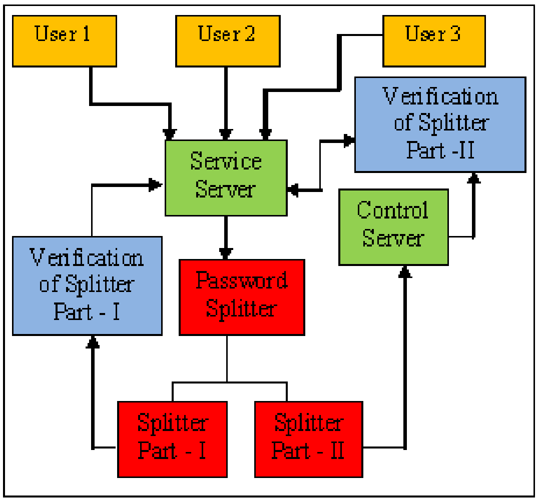


Fig 2: Generalized Two Server Architecture of a single control server with multiple service server

5. Integrated Quantum key distribution and Classical key

With QKDP implicit user authentication that confidentiality is only possible for legitimate users and mutual authentication is achieved only after secure communication using the session key start. The proposed three-party QKDPs are executed purely in the quantum channel, and this work does not consider errors caused by environmental noise. The proposed integrated QKDP and its classical security assumes that every participant shares a secret key with the TC in advance either by direct contact or by other ways. The integrated QKD and classical key model deployed in the two server password system are explained in the following phases.

Setup Phase

Let A and B be two users who would like to establish a session key. KTU is the secret key shared between TC and user U. Bit sequence in KTU is treated as the measuring bases between user U and the TC. If (KTV)i = 0, the basis D is chosen; otherwise, the basis R. Notice that (KTV)i denotes the ith bit of the secret key KTU.

Key Distribution Phase

The following describes the details of key distribution phase. Assume that the TC has been notified to start the 3AQKDP with A and B. TC and the users have to perform the 3AQKDP as follows:

Trusted Center

a. The TC generates a random number rTA and a session key SK. TC then computes h $(K_{TA}, r_{T}\oplus)$ ˆ $(SK||U_A|| U_B)$ for A and, similarly, rTB and R_{TB}=h (K_{TB}, r_{TB}) $(SK||U_B|| U_A)$ for B.

b. The TC creates the qubits, QTA, based on

$(r_{TA} || R_{TA})_i$

and $(K_{TA})_i$ for Alice where i = ,2,. . . .n and

$(r_{TA} || R_{TA})_i$

denotes the ith bit of the concatenation

$r_{TA} || R_{TA}$

- If $(r_{TA} || R_{TA})_i = 0$, $(K_{TA})i=0$, then $(Q_{TA})_i$ is 1\√2 (|0›+|1›)
- IF $(r_{TA}|| R_{TA})i=1$,$(K_{TA})i=0$, then $(Q_{TA})_i$ is 1\√2 (|0›-|1›).
- If$(r_{TA} ||R_{TA})i=0$,$(K_{TA})I =0$, $(K_{TA})i$ = 1, then $(Q_{TA})i$ is (|0›).
- If$(r_{TA} ||R_{TA})i=1$,$(K_{TA})I =1$, then $(Q_{TA})i$ is | 1)

TC then sends QTA to A. TC creates qubits QTB in the same way for B.

Users

a. A measures the received qubits QTA depending on KTA. If (KTA)i = 0, then the qubit is measured based on the basis D; otherwise, the basis R. Similarly, B measures the receiving qubits QTB depending on KTB.

b. Once A obtains the measuring results $r'_{TA}||R'_{TA}$, she then computes $SK'||U_A||U_B=h\ (K_{TA}, r'_{TA})\oplus R'_{TA'}$

The session key SK^1 can be obtained and the values UA and UB can be verified. Similarly, B gains $r'_{TB}||R'_{TB}$ and computes $SK''||U_B||U_A=h\ (K_{TB}, r'_{TB})$ $R'_{TB}\oplus$

Then, B obtains the session key SK00 and checks the correctness of UB and UA. In item a of TC, the hash value is used to encipher the sequence. Therefore, a recipient will not receive the same polarization qubits even if an identical session key is retransmitted. This also makes an eavesdropper not be able to perform offline guessing attacks to guess the bases over the quantum channel and, thus, the secret key, KTA (or KTB), can be repeatedly used.

In item b of Users, only A (or B), with the secret key KTA (or KTB), is able to obtain $SK'||U_A||U_B$ (or $SK''||U_B||U_A$) by measuring the qubits QTA (or QTB) and computing

$h\ (K_{TA}, r'_{TA})\oplus\hat{}\ R'_{TA}$ (or $h\ (K\oplus_{TB}, \hat{}\ r'_{TB})\ R'_{TB}$).

Hence, A (or B) alone can verify the correctness of the ID concatenation $U_A||U_B$ (or $U_B||U_A$)..

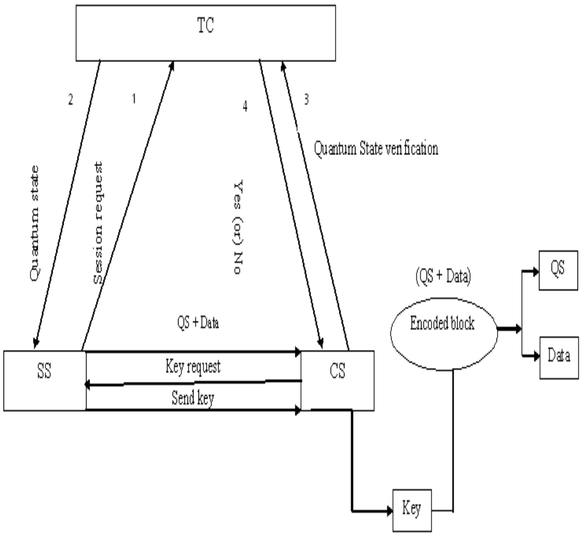


Fig 3: Process Flow diagram for Quantum based two server password authentication

Security Proof of QKDP

A new primitive, Unbiased-Chosen Basis (UCB) assumption, based on the no cloning theorem is also proposed to facilitate the proof. The UCB assumption describes that one can distinguish the polarization basis of an unknown quantum state with only a negligible probability.

Protocol Participant

A fixed nonempty set of legitimate participants and a TC are supposed to take part in 3QKDP. A participant and TC may have many instances correlated in distinct and concurrent executions of 3QKDP.

Long-term Secret Key

Every participant and TC share one secret key KTU, which is a sufficient long random binary string. TC maintains a table to store for every participant. Besides, U saves KTU as his long-term secret key.

Instance States

A client instance U accepts when it gains sufficient information to compute a session key SK. It should be noted that the state of acceptance only appears in client instances. Moreover, a client instance U can accept at any time and only accept once.

Session Identifier (SID) and Partner Identifier(PID)

The SID is used for a participant U to uniquely name his proceeding session. We define the SID for instance U in an execution of 3AQKDP. The PID names the participant with which a client instance affirms that it has just shared a session key SK. UA affirms that it has just shared SK with an instance of participant UB. It should be noted that the SID and PID are public and available to the adversary A.

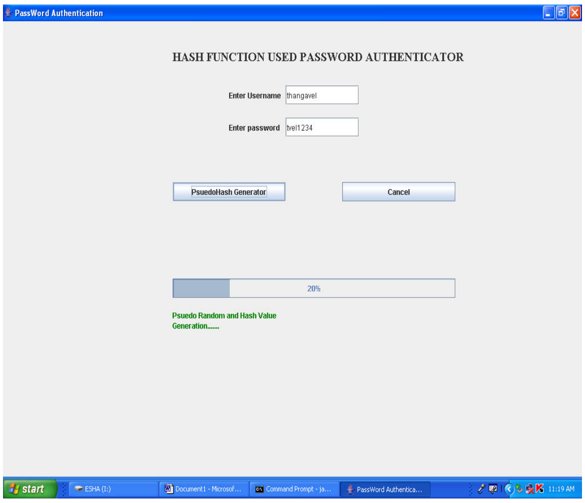
Adversary's Queries

The queries, Initiate query, Send query, Reveal query, Hash query, and Test query, represent the capabilities of adversary A.

6. Experimental Evaluation

In conduction of experimentation of hash pseudo random, a hash value is derived from the user's password, and the site domain name. Pass word Hash captures all user input to a password field and sends hash (pwd, dom) to the remote site, dom is derived from the domain name of the remote site. Hash is implemented using a Pseudo Random Function keyed by the password. Since the hash output is tailored to meet server password requirements, resulting hashed password is handled normally at the server with no server modifications required. Password Hash transparently converts a user's password into a domain-specific password. Password Hash automatically replaces the contents of these password fields with a one-way hash of the pair (password, domain-name). The site only sees a domain-specific hash of the password, as opposed to the password itself. A break-in at a low security site exposes password hashes rather than an actual password. Hash function used is public and can be computed on any machine which enables users to login to their web accounts from any machine in the world. Hashing is done using a Pseudo Random Function (PRF). Strong passwords are automatically generated. The same master key produces different passwords at many sites. Quickly upgrade passwords by bumping the site tag. Upgrade the master key without updating all sites at once. It supports different length passwords. It supports special requirements, such as digit and punctuation characters. All data is saved to the browser's secure password database

In our experimental implementation of two server system, a password is split into two random numbers.



PassWord Authentication

HASH FUNCTION USED PASSWORD AUTHENTICATOR

Enter Username: thangavel

Enter password: tvel1234

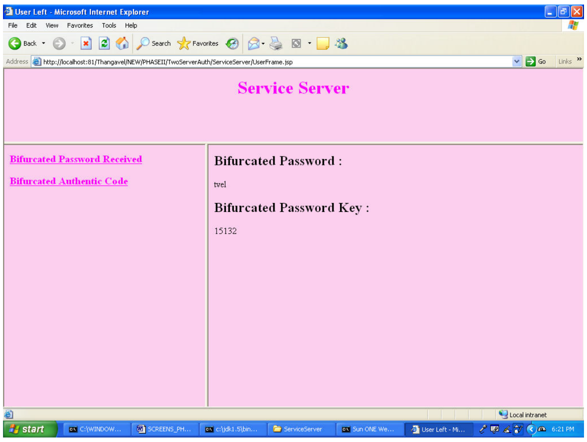
20%

Psuedo Random and Hash Value Generation.......

Fig 4: Pseudo Hash generation

Therefore, a user can use the same password to register to different service servers; they connect either to distinct control servers or to the same control server. This is a highly desirable feature since it makes the system user friendly. The big inconvenience in the traditional password systems is that a user has to memorize different passwords for different applications. The system has no compatibility problem with the single-server model. The user contacts only the service server but both the control and service servers are responsible for the authentication of the user. The user has a password which is transformed into two long secrets which are held by service server and control server. Both the system using their respective shares validate user during the login. The servers compute function to verify the user and finally a session key is being established between the user and service server for the confirmation of the user and the server. The service server which is an active adversary acts arbitrarily to uncover the passwords and could control the corruption of the password, the control server which is a passive adversary acts according to the protocol specification.

In the offline dictionary attacks, where the successful logins between the user and the server is recorded by the intruder and it tries the passwords in the dictionary against login transcripts and this is overcome in the system by control server as passive adversary and service server as active adversary (fig 4). In the system, the communication and the computations are more efficient. The user can use the same password to register to different service server, the service server connect either to distinct control servers or to the same control server. This is a highly desirable feature since it makes the system user friendly. The system could be adapted to any existing FTP and web applications that are available today by adding a control server to it where these are managed by the administrative domain.



User Left - Microsoft Internet Explorer

Address: http://localhost:81/Thangavel/NEW/PHASEII/TwoServerAuth/ServiceServer/UserFrame.jsp

Service Server

Bifurcated Password Received

Bifurcated Authentic Code

Bifurcated Password :

tvel

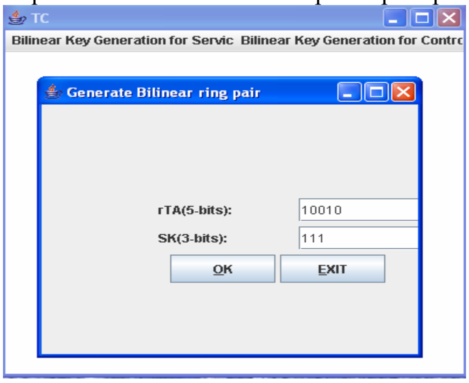
Bifurcated Password Key :

15132

Fig 5: Service server password authenticity

The generalization as well as the applications of the two-server password system well support the underlying security model, in the sense that the enterprise headquarter naturally assume adequate funds and strong security expertise and, therefore, affords and is capable of maintaining a highly trustworthy control server against both inside attackers and outside attackers. Without the concern of a single point of vulnerability, affiliating organizations that operate service servers are offloaded to some extent from strict security management, so they can dedicate their limited expertise and resources to their core competencies and to enhancing service provision to the users. From the perspective of users, they are able to assume the higher creditability of the enterprise while engaging in business with individual affiliating organizations.

In the implementation process of two server for password exchange between the servers combines classical key with quantum key model. It achieves key verification and user authentication. It preserves a long term secret key between the TC and each user. It measures EPR pairs and reconstructs TC and a participant after one QKDP execution. It detects the existence of passive attacks like eavesdropping. It resists replay and passive attacks. The three-party QKDPs, with implicit user authentication is designed. It executes three-party QKDPs purely in the quantum channel. Every participant shares a secret key with the TC in either by direct contact or by other ways. The three party QKDP allows explicit mutual authentication. The secret key pre-shared between the TC and a participant is long-term. The number of communication rounds is reduced to three. It integrates the advantages of both the classical and quantum cryptographies. Key distribution protocols facilitate sharing secret session keys between users on communication networks.(Fig 6) It provides secure communication on insecure public networks. A malicious attacker may derive the session key from the key distribution process. Designing secure key distribution protocols in security is a top priority. The three-party QKDP requires that the TC and each participant pre-share.


TC
Bilinear Key Generation for Servic
Bilinear Key Generation for Contrc
Generate Bilinear ring pair
OK
EXIT

rTA(5-bits): 10010

SK(3-bits): 111

Fig 6: Quantum Generator for Session keys

It provides secure communication on insecure public networks. A malicious attacker may derive the session key from the key distribution process. Designing secure key distribution protocols in security is a top priority. The three-party QKDP requires that the TC and each participant pre-share

7. Performance Result and Discussion on Quantum Hash Two Server Password Authentication System

The hash based multi-site pseudo random password mechanism considers N number of times that the user U might authenticate before re-registration is required. This suggests that high values of N are desirable. The host H has to store R hash function values at the server. This implies that to reduce the storage requirements, it is desirable to have a low value of R. However, N/2R is the average number of hash function computations that U has to do for every authentication session. Thus, it is desirable to have a high value of R. The parameter R therefore represents a tradeoff between computational requirements of the user U and the storage requirements of the host H. This implies that the value of N and R are best selected by the system administrator keeping in mind the system requirements. We believe that given the current state of storage technologies, the storage requirement is significantly less important than the computational requirement. Major improvement over the previous cryptographic method is the significant reduction in computational requirements per authentication session and increase in the number of logins before re-initialization.

Regarding the computation evaluation the host verifies the proposed hash password sent by user by computing just a single hash function and one comparison with the stored last one time password. For the investigation of communication factor the host sends the user a hash value and an integer t. The user returns only a single hash value. The resultant of the proposed hash based pseudo random password authentication and cryptographic password authentication are listed in the below Table 1.

Table 1: Effectiveness of proposed hash based pseudorandom password authentication over existing cryptographic password authenticity

Technique	***Resistance to eavesdropping***	***Web Browser Compatibility***	***Web browser Compatibility Number of rounds for authentication***	***Computational Efficiency***	***Storage Capacity***	***Communication effectiveness***
Existing Cryptographic password authentication	Feasible	False	Low	Low	High	False
Ptoposed hash based pseudo random password authentication	Highly feasible	True	High	High	Low	True

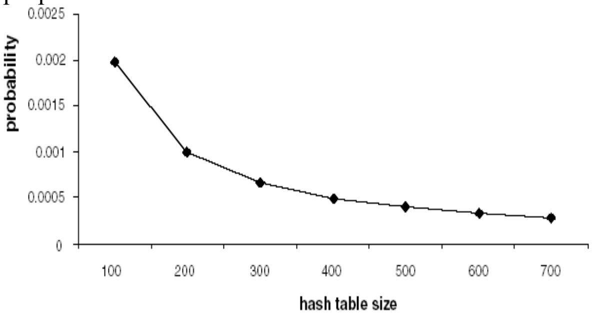
7.1 Performance Measure on Two Server Authentication

The exponentiations dominate each party's computation overhead, the two server password authentication system only count the number of exponentiations as the computation performance. The digits before "/" denote the total number of exponentiations performed by each party, and the digits following "/" denote the number of exponentiations that can be computed offline. One round is a one-way transmission of messages. The proposed two protocols demonstrate performance quite efficient in terms of both computation and communication to all parties. Take U, for example, it needs to calculate 3 and 4 exponentiations in the two protocols, respectively, and 2 of them can be performed offline. This means U only computes 1 and 2 exponentiations in real time in the respective protocols, the communication overhead for U is particularly low in terms of both bits and rounds.

Table 2: Performance measure on Two server and Single server password authentication scheme

Scheme	Time of Authenticity rate	Success
	(milliseconds)	%
Two server password authentication	10	96
Single server	8	87

The table2 listed above indicates the computation performance in terms of time and success rate (number of rounds) of the two server password authentication and single server authentication. The performance graph 1 show the probability of success rate against the hash table size in the two server hash password authentication system proposed here.



Graph 1: Probability of success rate Vs Hash table size for two server password authentication

7.2 Performance Issue on classical and quantum key on two server

In the security proofs, the capability of an adversary is modeled by queries, which also represent the possible attacks performed by an adversary. However, since the online guessing attack in which an adversary guesses the possible secret and judges the correctness of the guess by the execution result of the protocols cannot be avoided in existing key distribution protocols, as no proper queries have been adopted to model this attack in existing security proofs. An online guessing attack is not modeled in the security proofs of older systems. The online guessing attack can occur when an adversary performs an intercept-resend attack on one qubit at a time (by say starting from the first qubit) over the qubit sequence sent from TC. The adversary intercepts the qubit sequence and measures the first qubit using an arbitrary basis. Then, the adversary produces a qubit according to the measurement result to replace the first qubit of the intercepted sequence, and then resends the new qubit sequence to the participant.

The adversary then observes the participant reaction. In the case of a negative reaction (25 percent probability), the adversary immediately knows the correct basis; otherwise, the adversary has to repeat the process on the same bit in the next executions of protocols until a sufficient degree of certainty for the basis of this qubit can be acquired. The adversary then proceeds to the next qubit following the same strategy. After a number of rounds, the adversary may know with a high probability all proper basis positions and the respective key. Table 3 Shows the performance improvement of proposed Quantum and classical key password authentication model with other tradition cryptographic techniques.

Table 3: Comparison of Proposed Quantum and classical to individualized classical and quantum key models

Performance metrics	*Proposed Quantum Key & classical*	*Quantum key Model*	*Classical Key Model*
Pre-shared Secret key	Longer Duration	Sampling pair instances	Longer Duration
Communication Round	2	5	3
Quantum Channel	Yes	Yes	No
Clock Synchronization	No	No	No
Vulnerable to Passive Attack	No	No	No
Security Proof	Yes	No	No

7.3 Discussions

With two-server password system, single point of vulnerability, is totally eliminated. Without compromising both servers, no attacker can find user passwords through offline dictionary attacks. The control server being isolated from the public, the chance for it being attacked is substantially minimized, thereby increasing the security of the overall system. The system is also resilient to offline dictionary attacks by outside attackers. This allows users to use easy to remember passwords and still have strong authentication and key exchange. The system has no compatibility problem with the single-server model. The generalization of the two-server password system well supports the underlying security model. In reality, adversaries take on a variety of forms and no security measures and precautions can guarantee that a system will never be penetrated. By avoiding a single point of vulnerability, it gives a system more time to react to attacks. The password-based authentication and key exchange system that is built upon a novel two-server model, where only one server communicates to users while the other server stays transparent to the public. Compared with previous solutions, our system possesses many advantages, such as the elimination of a single point of vulnerability, avoidance of PKI, and high efficiency.

Among classical three-party key distribution protocols focuses on the low bounds of communication rounds of three-party key distribution protocols, such as the low bound of timestamp-based protocols and the low bound of nonce-based protocols. Therefore, this project evaluates the communication rounds with the proposed protocol. The three party QKDP allows explicit mutual authentication is chosen for comparison. The three-party QKDP avoids passive and replay attacks due to the quantum phenomena. Pre-shared key pair is used between the TC and participants to prevent man-in-the-middle attacks. However, not only must participants perform public discussions to verify the correctness of the session key, but the pre-shared pairs must be reconstructed for each session. The classical three-party key distribution protocols utilize challenge-response mechanisms or timestamps to prevent replay attacks. However, challenge-response mechanisms require at least two communication rounds between the TC and participants, and clock synchronization is impractical. Furthermore, classical cryptography cannot detect passive attacks such as eavesdropping. By integrating the advantages of both classical and quantum cryptographies, the proposed model avoid man-in-the-middle, passive, and replay attacks. Furthermore, since the challenge-response mechanism is no longer necessary, the number of communication rounds is reduced to three, the same as the low bound in the timestamp-based protocol, and one fewer than the low bound of the challenge-response protocol.

8. Conclusion

The hash password system presented provably secure hash functions based password authentication scheme. This construction provides features such as both the block size of the hash function and the output size are completely scalable. The password hashing method is extremely simple, rather than send the user's clear text password to a remote site, it sends a hash value derived from the user's password, and the site domain name. The developed two-server password authentication architecture has control server and service server. The control server is controlled by a passive adversary while the service server is controlled by an active adversary. A single point of vulnerability, as in the existing password systems, is totally eliminated. Without compromising both servers, no attacker can find user passwords through offline dictionary attacks. The control server being isolated from the public, the chance for it being attacked is substantially minimized, thereby increasing the security of the overall system. The system has no compatibility problem with the single-server model.

The two server authentication utilizes the advantages of combining classical key with quantum key model to improve the performance of password sharing between the control server and service server. Compared with classical three-party key distribution protocols, the proposed one easily resists replay and passive attacks. Compared with other QKDPs, the proposed schemes efficiently achieve key verification and user authentication and preserve a long term secret key between the TC and each user. The keys are stored and managed within key stores, placed in nodes, and not within QKD devices or within the machines running endpoint secure applications. This design choice allows to manage keys over a dedicated global network (the network of secrets) composed of key stores linked together with classical channels. The network of secrets is by essence a classical network as being called as QKD. The underlying key generation mechanism, responsible for filling the key stores, is quantum key distribution. The proposed integrated key model had fewer communication rounds than other protocols. By combining the advantages of classical cryptography with quantum cryptography, this work presents a new direction in designing QKDPs.

REFERENCES

[1] J. Brainard, A. Juels, B. Kaliski, and M. Szydlo, "A New Two Server Approach for Authentication with Short Secrets," Proc. USENIX Security Symp., 2003.

[2] S. Bellovin and M. Merritt, "Encrypted Key Exchange: Password Based Protocols Secure against Dictionary Attacks," Proc. IEEE Symp. Research in Security and Privacy, pp. 72-84, 1992.

[3] Z. Zhao, Z.Dong, Y.Wang, "Security Analysis of a Password-based Authentication Protocol Proposed to IEEE 1363, " Theoretical Computer Science, Vol.352, No.1, PP.280-287, 2006.

[4] Kumar Mangipudi and Rajendra Katti, "A Hash-based Strong Password Authentication Protocol with User Anonymity", International Journal of Network Security, Vol.2, No.3, PP.205–209, May 2006 .

[5] M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols," Proc. ACM Computer and Comm. Security, pp. 62-73, 1993.

[6] W. Ford and B.S. Kaliski Jr., "Server-Assisted Generation of a Strong Secret from a Password," Proc. IEEE Ninth Int'l Workshop Enabling Technologies, 2000.

[7] D.P. Jablon, "Password Authentication Using Multiple Servers," RSA Security Conf., pp. 344-360, 2001.

[8] P. Mackenzie, T. Shrimpton, and M. Jakobsson, "Threshold Password-Authenticated Key Exchange," Proc. Advances in Cryptology (Eurocrypt '02), pp. 385-400, 2002.

[9] D. Gottesman and H.-K. Lo, "Proof of Security of Quantum Key Distribution with Two-Way Classical Communications," IEEE Trans. Information Theory, vol. 49, p. 457, 2003.

[10] H.A. Wen, T.F. Lee, and T. Hwang, "A Provably Secure Three- Party Password-Based Authenticated Key Exchange Protocol Using Weil Pairing," IEE Proc. Comm., vol. 152, no. 2, pp. 138-143, 2005.

[11] J. Nam, S. Cho, S. Kim, and D. Won, “Simple and Efficient Group Key Agreement Based on Factoring,” Proc. Int’l Conf. Computational Science and Its Applications (ICCSA ’04), pp. 645-654, 2004.

[12] B. Neuman and T. Ts’o, “Kerberos: An Authentication Service for Computer Networks,” IEEE Comm., vol. 32, no. 9, pp. 33-38, 1994.

[13] Abdalla M., Catalano D., Chevalier C., and Pointcheval D., “Effcient Two-Party Password Based Key Exchange Protocol in the UC Framework”, Springer-verlag Berlin, pp 335-351, 2008.

[14] Tzonelih Hwang, Kuo-chang Lee and Chuan – Mingli “Provably secure Three party Authenticated Quantum Key Distribution Protocols” IEEE Transactions on Dependable and Secure computing Vol.4, No.1, PP.71-80,2007.

[15] M. Bellare, D. Pointcheval, and P. Rogaway, “Authenticated Key Exchange Secure Against Dictionary Attacks,” Advances in Cryptology (Eurocrypt ’00), pp. 139-155, 2000

[16] C.H. Bennett and G. Brassard, “Quantum Cryptography: Public Key Distribution and Coin Tossing,” Proc. IEEE Int’l Conf. Computers, Systems, and Signal Processing, pp. 175-179, 1984.

[17] C.H. Bennett, “Quantum Cryptography Using any Two Nonorthogonal States,” Physical Rev. Letters, vol. 68, no. 3121, 1992.

[18] A.K. Ekert, “Quantum Cryptography Based on Bell’s Theorem,” Physical Rev Letters, vol. 67, no. 661, 1991

[19] N. Chou, R. Ledesma, Y. Teraguchi, and J. Mitchell, “ Client-side defense against web based identity theft “, In Proceedings of Network and Distributed Systems Security (NDSS), 2004.

[20] J. A. Halderman, B.Waters, and E. Felten “A convenient method for securely managing passwords” To appear in Proceedings of the 14th International World Wide Web Conference (WWW 2005), 2005.

[21] F. Hao, P. Zielinski, “A 2-round anonymous veto protocol,” Proceedings of the 14th International Workshop on Security Protocols, SPW’06, Cambridge, UK, May 2006.

[22] Muxiang Zhang, “Analysis of the SPEKE password-authenticated key exchange protocol,” IEEE Communications Letters, Vol. 8, No. 1, pp. 63-65, January 2004.

T. Thangavel received he Bsc degree in Computer Science (Bharathiyar University) in 1991 and the Msc degree in computer science Bharathidasan University) in 1993 and the Mphil degree in Computer Science (Bharathidasan University) in 2003. He is pursuing the PhD degree in department of science and humanities (Anna University). He is working as an Assistant Professor in MCA department at K.S.Rangasamy College of Technology, Tiruchengode

[2]**Dr. A. Krishnan** received his Ph.D degree in Electrical Engineering from IIT, Kanpur. He has been in the field of technical teaching and research for more than four decades at Government College of Technology and Coimbatore Institute of Technology, Tamilnadu, India. From 1994 to 1997, he was an Associate Professor in Electrical Engineering at University Pertanian Malaysia (UPM), Malaysia. He is now working as an Academic Dean at K.S.Rangasamy College of Technology, Tiruchengode and research guide at Anna University Chennai. His research interest includes Control system, Digital Filters, Power Electronics, Digital Signal processing, Communication Networks. He has been published more than 180 technical papers at various National/ International Conference and journals. Dr. Krishnan is a senior member of IEEE, Life fellow Institution of Engineers (India), IETE (India) and Computer Society of India.