Key Establishment and Authentication of User by Specialized Digital Certificate for Secure Communications

DASARI REVENTH RAJ$^{\#1}$, VASUPALLI MAHESH$^{\#2}$, Y.RAMESH KUMAR$^{\#3}$

[1] Final Year M.Tech Student, [2] Associate Professor, [3] Associate Professor & Head of Department
[1,2,3] Department of Computer Science and Engineering
[1,2,3] Avanthi Institute of Engineering and Technology, An NBA Accredited College, Affiliated to JNTU Kakinada,
Cherukupally(V),Bhogapuram(M), Vizianagaram(Dist), Andhra Pradesh State, India.
[1] reventh.mtech@gmail.com, [2] mahi.vasupalli@gmail.com, [3] javaramesh143@gmail.com

Abstract - **In order to provide public key authentication to a user in Public-key Infrastructure (PKI), we were widely using the Public-key digital certificate. But this certificate is not validated to authenticate user under security aspects. So, in order to overcome this aspect, we were proposing Specialized Digital Certificate which avails the authentication of user and key agreement for secured transmission of data. It consists of users general information such as digital driver's license, digital birth certificate etc. and trusted security authorities signed digital signature of public information. Also, it doesn't have user's public key. As user, doesn't contain public key and private keys. So the methodology of key management what we were going to use in Specialized Digital Certificate (SDC) is much simpler than using public -key digital certificate. The digital signature which is generated by SDC will be never revealed to any verifier as it is used as a secret token by user. Since the owner proves to the verifier that he authorizes the certificated by responding to verifiers challenge. Based on this concept, we considering both Discrete Logarithms (DL) based and Integer Factoring (IF) based protocols which suits for authentication to user and secret key generation and establishment.**

Index Terms – Public-key digital certificate, user-authentication, key-management.

I. INTRODUCTION

As the digital certificate is nothing but combining a statement and its digital signature. Generally, statement will contain the user's public key obviously with other information. Till now, signer of this digital signature is trusted third-party Certificate Authority (CA). “X.509 public – key digital certificate” is the mostly used digital certificate to provide authentication for user's public-key contained in the certificate. If the user is able to reveal that he had relevant information of private key corresponding to public-key which is provided in the above certificate, then only he will be authenticated. So, only the public-key digital certificate can’t consider authenticating user. Because public-key digital certificate contains only public information and it will easily record and play back once it was revealed to verifier.

In this research paper, we were entitling an new approach which makes an user to be authenticated and with his partner a shared secret session key will be established in the general form of digital Certificates, such as digital driver´s license, a digital birth certificate of digital ID etc. We consider this kind of digital certificate as a Specialized Digital Certificate (SDC). A SDC contains public information related to the users and CA signed digital signature of this public information. However in this SDC technique, public information does not contain any user´s public key.

As the user will not have any public and private key pair, this type of digital certificate technique will be much easier to manage than the well renowned X.509 public-key digital certificates. The digital certificate of SDC technique is considered as secret token of each user. The owner of this SDC never reveals signature to verifier in plain text. Instead of it, the owner will computes a challenge to the verifiers to prove that he has sufficient knowledge regarding the digital signature. So by using this SDC technique, we can provide user authentication in a digital world. Including with it, a secret session key will be established between the verifier and the certificate owner during this interaction.

Mainly there are 3 entities in a digital certificate application. Those are as follows:

a) Certificate Authority (CA): It is the person or organizations which will digitally signs a statement with its private key. In PKI applications, the X.509 public-key digital certificate will have a statement, which includes the user´s public key, and a digital signature of statement. The main difference between the proposed technique and the existing technique is that in proposed technique, the public information does not contain any user´s public key.

b) Verifier: The verifier is the people who will challenges the owner of SDC and validates the answer using the owner´s public information and CA´s public key.

c) Owner of SDC: The owner of this SDC is the person who receives SDC from a trusted CA over a secure channel. The owner needs to compute a valid "answer" in response to the verifier ´s challenged "question" in order to be authenticated and establish a secret session key.

Generally in paper-world user identification applications, the responsible for trusted authority is for issuing identification card with their information like user name and photograph on card. Every user will be successfully identifies if they have owns their legitimate like “paper certificate” and matches with photograph. The tamper-resistant tech makes the identification cards to be more secured i.e. forgery can't be takes place. Having paper certificate, is one of factor in process of authentication. In this technique, our aim is to show a similar path in electronic-world applications. We consider it is as Specialized Digital Certificate (SDC). It contains public data of user and trusted certificate authority issued digital signature of the public information. The certificate will be never exposed to the verifier. Such that, SDC becomes security factor which can be used for user authentication.

The remaining part of this paper is considered as follows. In the beneath section, we shows an overview of related data of work. After that section, we consider preliminaries and also mention the discrete Logarithm (DL) based user authentication and key establishment protocol using SDC. After that section, we consider integer factoring (IF) based user authentication and key establishment protocol. Hence we conclude after that section.

II. RELATED WORK

Authentication of user and establishment of key are the basic fundamental services in secured communication. A prolonged research has been conducted in both fields. Whatever, unlike the (NEW) which we consider in this paper, most methods in literature rely on the digital certificates of public-key for providing user authentication and key establishment [3]-[5].

The basic digital signature provides authentication of given message to the receiver. But these approaches were violating the privacy of the signer's. A receiver who is malicious can violate the signer's privacy. They can reveal the sender's digital signature to any other instances without the liable concern. Subsequently, the signer's public-key can be accessed by anyone and the digital signature is validated. In the year of 1989, the two research persons Chaum and Antwerpen [6] introduces the concept of signature which is undeniable, which will enables the signer to have a complete control over their signature. Undeniable signature verification requires participation of signer's message. So, this fixture prevents the undesirable verifiers from signature validation. The crucial problem of the undeniable signature is signer needs to make the verifier authentication before helping the verifier to validate the signature which is undeniable. Some recent research works data can be found in [7], [8].

DVS (Designated Verifier Signature) was introduced in [9], and also in [10] independently, both in 1996. It provides authentication of a that a valid one can be generated by "real" signer or by the designated verifier. By this, it is differentiated from a basic digital signature in two points, (1) as the designated verifier knows that they did not generate the DVS by themselves, the designated verifier were convinced that, traditional signature can be verified by real signer. However, like the traditional one, which can be verified by any one of verifiers. Any third-party member can't determine the real signer of DVS even with knowledge of private key. (2) DVS provides authentication of message without traditional digital certificate which property is non-repudiator. It can replace the traditional one in most applications and provides services with deniability.

In [9], Designated Verifier Signature based on a signature which is non-interactive undeniable scheme with a commitment which is trap-door was proposed, but this scheme becomes inefficient computationally. DVS might be established by giving the number of signers in a ring signature to two as proposed in [11], [12]. So DVS, which is based on ring signatures can't provide verifier properties which were strong designated. In [13], a DL-based DVS scheme which is based on the combination of Schnorr signature [14] and Zheng Signature [15] were proposed. Simply it is a pairing-based variant of [11]. In Latest, these DVS schemes were based on any bilinear map were proposed [16].

Mainly, the concept of Universal Designated Verifier Signature (UDVS) was proposed in [17]. It is an ordinary digital signature which converts the signature into a DVS of any designated verifier depends upon his choice. The construction of this scheme (DVSBM) was based on a bilinear map. They were three new UDVS constructions based on Schnorr [14] and RSA signatures [17] were proposed in [19]. Also, the ElGamal-based UDVS were proposed in [16]. The other related research on the DVS and UDVS were found in [20]-[22].

Relevant to our proposed scheme, there were three entities in each UDVS application: the Certified Authority (CA), digital signature owner and designated verifier. So, in UDVS the user needs to makes the conversion in digital signature into DVS non-interactively inorder to authenticates the information. But in our proposed scheme, the digital certificate owner interacts with verifier to prove the knowledge of digital certificate and to be authenticated by the user.

Our proposed aspect is closely related to the cryptography which is ID-based [23]. In an cryptographic algorithms which were ID based, each user must be register at a private key generator (PKG) and which identifies himself before getting into the network. Once the users were accepted, the PKG will generate a private key for user. Here the user's identity becomes the corresponding public-key. By this way, to verify a digital signature of a message, sender sends an encrypted message to receiver, user needs to know the "identity" of his communication partner and the public key of PKG, which were extremely used in cases like wireless communication where public keys pre-distribution is infeasible. Whatever, an ID-base cryptographic algorithm, assumes that each user already knows identity of their communication partner? According to this assumption, there is no need or not having any feasible ways to authenticate the identity. This is the main advantage of cryptography which was ID-based. Because of this assumption, ID cryptography were only limited to applications whose communication entities know each other prior to communication. But in our considering scheme, the user is not necessary to know any information of his/her communication partner. The public information of this (NEW) scheme, such as user's identity, can be transformed and verified by each communication part. Furthermore, it is used to make authentication for each other. In another instance, our proposed data supports general PKI applications, such as Internet, e-commerce, whose communication entities doesn't need to know each other prior to communication. Our proposed scheme is based on combination of conventional digital signature scheme and the well-known (generalized) Diffie-Hellman assumption [24], [25].

III- DL- BASED PROTOCOL

A. Preliminaries:

One of the advantages of having a paper certificate over public-key digital certificate is that a paper certificate cannot be easily cloned or Modified, but a feasibility with public-key digital certificate is it can be recorded and played back easily.

Here there is a provision that there is no need of revealing the digital signature of the SDC in plain text to the verifier by the SDC owner. Instead, the owner proves that he has knowledge of the digital signature by responding to the verifiers challenge. The knowledge of SDC owner provides user authentication. For this the following requirements should be satisfied.

1) **Unforgeability**: Only a person who knows the digital signature of the SDC can generate a response which is only valid

2) **One-wayness**: Basing on interaction no person can derive the certificate's digital signature.

3) **Nontransferrability:** There is no chance of creating impersonation of the user.

Our protocol was completely based on DL-based digital signature and Diffie-Hellman assumption.(DHA) [24].

B. Review of ElGamal Digital Signature

In the ElGamal scheme [26], a large prime p and a generator g in the order of p−1 are left to be shared by all users in assumed mode. The signer selects a random private key x ∈[1, p − 2] and makes the computation of corresponding public key $y = g^x \textit{ mod } p$. First, signer randomly selects a secret parameter $k \in [1, p-1]$ with $gcd(k, p-1) = 1$ and computes $r = g^k \textit{ mod } p$. Then, s is solved as the Signer's secrets, x and k were known as

$$m = ks + rx \bmod p - 1, \quad (1)$$

Where m, represents the message digest for the message m′. (r, s) is defined as the digital signature of the message m′. The signature (r, s) can be verified by checking whether the equation

$$g^m = y^r r^s \bmod p, \quad (2)$$

holds true.

In an ElGamal signature scheme, the parameter r of the signature can be computed off-line as $r = g^k \textit{ mod } p$. The signature component s is in online which was computed. The readers can refer to [27] for more discussion on the design of DL based signature schemes. Without generality losses, we represent the signing equation in generalized format for all DL-based signature schemes as $ax = bk + c \textit{ mod } p - 1$ where (a, b, c) are three parameters from the set of values (m, r, s). More specifically, each parameter can be a mathematical combination of (m, r, s). For example, the parameter a can be m, r or s. The verification equation is determined accordingly as $y^a = r^b g^c$ mod p. There are 18 generalized ElGamal-type signature variants [27].

In the following discussion, we use the original ElGamal signature as an example to present our proposed protocol.

C. Diffie – Hellman Assumption (DHA)

Consider A and B have their own private keys, x_A and x_B, and their respective public keys, $y_A = g^{xA} \textit{ mod }_p$ and $yB = g^{xB} \textit{ mod } p$, respectively, where large prime integer is p and primitive element of the multiplicative group modulo p is g. Only A and B can stay a shared secret $K_{A,B} = y^xB^A = y^xA^B = K_{B,A} \textit{ mod } p$. DHA makes an assumption that it's infeasible to determines $K_{A,B}$ not knowing the private key x_A or x_B from the having public-key y_A or y_B which were equivalent to solving the problem on discrete logarithm.

D. User Authentication and Key Establishment Protocol

I) Registration at CA: Let A be the certificate owner and B be the verifier. A needs to register at a CA to obtain a SDC. The CA generates an ElGamal signature (r_A, s_A) for user A's statement m'_A according to equation (1), where m_A is the message digest of the statement m'_A. Since the signature component r_A is a random integer and does not depend on m_A, it does not need to be kept secret. So, the signature component s_A is a function of the statement. Each owner needs maintains it as secret from the verifier in the authentication process. Our user authentication and key establishment protocol is illustrated in Fig. 1.

II) Protocol: The key establishment protocol and authentication contains the following four steps:

1) The user A passes his user information m'_A and parameters (r_A,S_A) to the verifier B, where $S_A = r^sA^A \textit{ mod } p$.

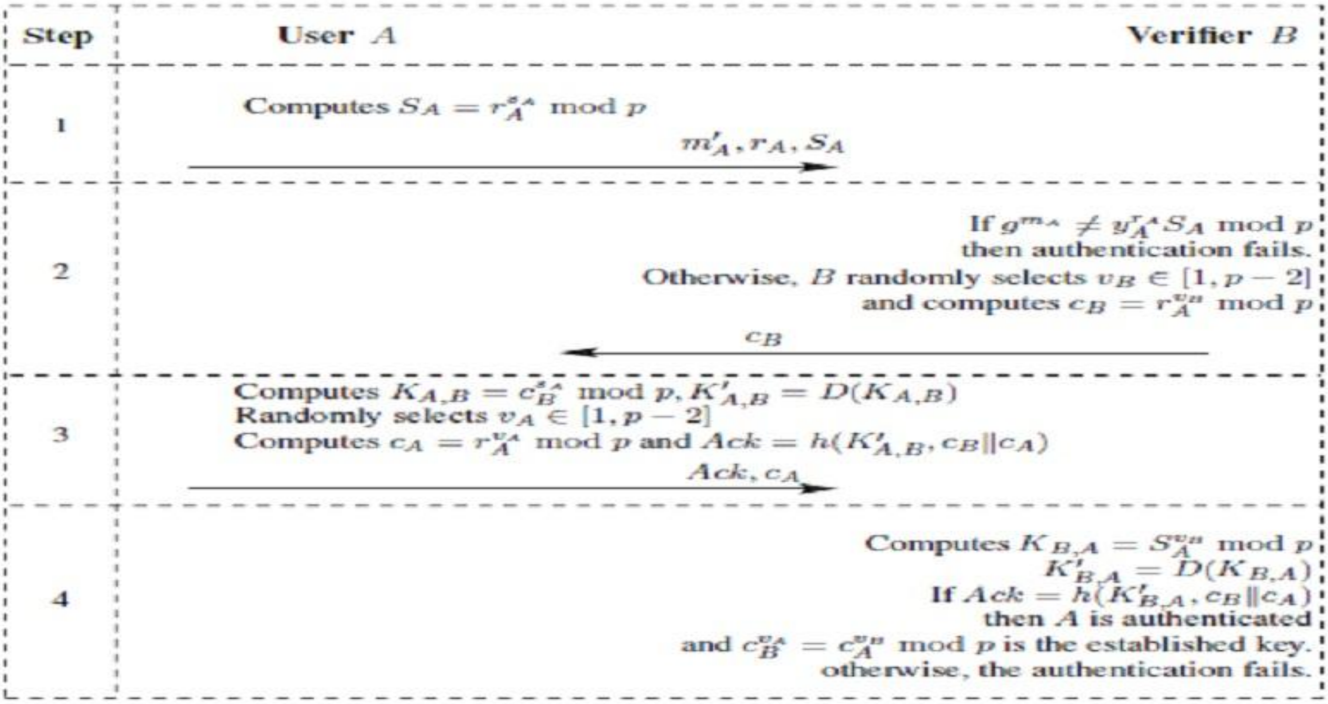
2) After receiving m'_A and (r_A,S_A), the verifier checks whether

$$g^{mA} = y^{rA} S_A \bmod p, \quad (3)$$

where y is the public key of the CA. If the holds true, the verifier B first randomly selects an integer $v_B \in [1, p - 2]$, then computes a challenge $c_B = r^vA^B$ mod p and send c_B to the user A. Otherwise, the user authentication fails and the protocol is stopped.

3) The user A first uses his secret s_A to compute the Diffie-Hellman secret key $K_{A,B} = c^sB^A \textit{ mod } p$, $K'_{A,B} = D(K_{A,B})$, where $D(K_{A,B})$ represents a key derivation procedure with $K_{A,B}$ as an input. Then user A randomly selects an integer v_A ∈[1, p − 2], computes $c_A = r^vA^A$ mod p and the response $Ack = h(K'_{A,B}, c_B||c_A)$, where $h(K'_{A,B}, c_B||c_A)$ represents a one-way keyed-hash function under the key $K'_{A,B}$. The user A sends Ack and c_A back to B.

4) After receiving the Ack and c_A from the user A, the verifier B uses his secret v_B to compute the Diffie-Hellman shared secret key $K_{B,A} = S^vA^B \textit{ mod } p$, $K'_{B,A} = D(K_{B,A})$, and checks whether $h(K'_{B,A}, c_B||c_A) = Ack$ is true. If this verification is successful, the certificate owner A is authenticated by the verifier B and a onetime secret session key $c^vB^A = r^vA^{AvB} = c^vA^B \textit{ mod } p$ is shared between A and B. This shared key can provide perfect forward security. In order to make verifier to successfully authenticated in our protocol, the


Step
User A
Verifier B
1
Computes $S_A = r_A^{s_A} \bmod p$
m'_A, r_A, S_A
2
If $g^{m_A} \neq y_A^{r_A} S_A \bmod p$
then authentication fails.
Otherwise, B randomly selects $v_B \in [1, p-2]$
and computes $c_B = r_A^{v_B} \bmod p$
c_B
3
Computes $K_{A,B} = c_B^{s_A} \bmod p$, $K'_{A,B} = D(K_{A,B})$
Randomly selects $v_A \in [1, p-2]$
Computes $c_A = r_A^{v_A} \bmod p$ and $Ack = h(K'_{A,B}, c_B||c_A)$
Ack, c_A
4
Computes $K_{B,A} = S_A^{v_B} \bmod p$
$K'_{B,A} = D(K_{B,A})$
If $Ack = h(K'_{B,A}, c_B||c_A)$
then A is authenticated
and $c_B^{v_A} = c_A^{v_B} \bmod p$ is the established key.
otherwise, the authentication fails.

owner of certificate needs to compute and send a valid pair (r_A,S_A) and Ack to the verifier in steps 1) and 3). The parameters (r_A, S_A) need to satisfy

$$g^{mA} = y^{rA}S_A \text{ mod} p.$$

This pair of integers can be easily solved by anyone. However, we want to show that only the certificate owner 'A' who knows the secret exponent of S_A can compute a valid Ack. This is because the verifier B can compute the one-time secret key $K_{B,A}$ used in generating the Ack as $K_{B,A} = S^vA^B= r^sA^{AvB} mod\ p$. According to the DHA, the certificate owner A who knows the secret exponent of S_A can also compute $K_{A,B}$ as $K_{A,B} = c^sB^A = r_{sA}A^{vB} = K_{B,A}\ mod\ p$. Thus, the certificate owner interacts with the verifier and successfully authenticated.

Remark 1: As we have discussed previously, a valid S_A can be solved by anyone, including the verifier. Thus, technically, S_A does not need to be transmitted in step 2). However, if the proven sends S_A in step 2), it can help the verifier to terminate the protocol immediately once an invalid S_A is detected.

E. Security Analysis and Discussion

In this scenario, we will consider the security of the proposed user authentication and key establishment protocol for the unforgeability, one-wayness and nontransferability.

a) Unforgeability: For the forgery attack to perform, the attacker needs to present a valid pair (r_A, S_A) in step 1) and the corresponding Ack in step 3) in order to impersonate the certificate owner successfully. A valid pair (r_A, s_A) alone in step 1) cannot be used to authenticate the certificate owner since this pair of parameters can be solved easily by the attacker from equation (3). However, it is computationally infeasible for the attacker to find the discrete logarithm of S_A because of the security signature scheme of ElGamal. So, it is computationally infeasible for the attacker to get a pair (r_A, s_A)to satisfy $g^{mA} = y^{rA}r^sA^A$ mod p. Due to the DHA, without knowing the secret exponent of s_A it would be infeasible for the attacker to compute $K_{A,B}$ and forge a valid Ack in step 3). On the other hand, the certificate owner obtains the secret exponent of S_A from CA during the registration and the certificate owner can be authenticated in step 3) successfully. At a glance, the unforgeability security of our proposed protocol is provided through combination of the security of the ElGamal signature scheme and the DHA. Therefore, the proposed key establishment and user authentication protocol is secure against forgery attacks.

b) One-wayness: In step 1), the certificate owner presents S_A to the verifier. The computation of secret s_A from s_A is infeasible since discrete logarithm problem computation is s_A from the S_A. Again, in step 3), the certificate owner uses the secret s_A to compute the Diffie-Hellman key $K_{A,B}$. Although the verifier knows the Diffie-Hellman key $K_{A,B}$; but due to the DHA, the verifier cannot obtain the secret s_A Therefore, our proposed protocol satisfies the one wayness property.

c) Nontransferability : Due to the DHA, a valid response Ack can only be generated by a certificate owner who knows the secret digital signature component s_A such that $r^sA^A = S_A$ mod p, or by a verifier who knows the random secret of a verifier.s selected random challenge. Since random challenge each time selects by the verifier, the response is only valid for a one-time authentication.

Since the digital signature of a GDC is never passed to the verifier, the verifier cannot pass the complete GDC to any third party. In our protocol, there is no privacy intrusion. Therefore, a valid response Ack cannot be transferred into a response of another verifier's challenge.

Our protocol enables a certificate owner to be authenticated and two one-time shared secret keys $K_{A,B}$ and $c^vB^A= r^vA^{AvB}= c^vA^B\ mod\ p$ be established between A the certificate owner, who knows s_A such that $r^sA^A = c^vA^B$ mod p and the verifier B through the authentication protocol. The former is used to generate the Ack, and the latter is established shared secret key between A and B. In addition, it enables the owner to send verifiers confirmation Ack. *As* the Diffie-Hellman secret shared key can be generated by either A or B, the certificate owner A can deny participating in the protocol.

Remark 2: In the original DHA, it is assumed that the generator $g$is a primitive element of the multiplicative group modulo p; while the parameter $r_A = g^k$ mod p in Theorem 1 is not necessarily a generator. However, we can ensure that r_A is a primitive element of the multiplicative group modulo p by requiring gcd (k,p-1) = *1*. Particularly, when $p= 2p'+1$ is a safe prime, where p' is also a prime, we can ensures r_A is a primitive element of the multiplicative group modulo $p$if $k$is an odd number.

Remark 3: Similar to the ID-based cryptographic algorithms, our proposed protocol also has the key insurance problem, i.e. the CA knows the users one-time secret session key which were shared between them. Few of cryptographic algorithms have been proposed to solve the key escrow problem of the ID based signature (IBS) while enjoying the benefits of the IBS, such as certificate less digital signature (CDS) .

IV. IF- BASED PROTOCOL

The scenario we are discussing to propose an IF-based user authentication and key establishment protocol. It is a combination of an online/off-line digital signature and a generalized Diffie- Hellman assumption (GDHA) [25].

A. Review of On-line/Off-line Digital Signature

We will review the trapdoor hash families and the online/off line signature scheme based on the trapdoor hash families.

A trapdoor hash family, introduced in [31] and officially defined in [30], having a pair of (I,H), where I is a probabilistic polynomial-time key generation algorithm, and H is randomized hash family. generates a pair (HK,TK), where HK is public hash key, and TK is its associated private trapdoor key. A trapdoor hash function in h is a hash function with a trapdoor secret. It is represented as $h_{HK}(m, s)$, where m is a message and s is an auxiliary random number. A trapdoor hash function must satisfy the following three requirements:

i) **Efficiency**: Considering a hash key HK and a pair (m,s), $h_{HK}(m,s)$ is computable in polynomial time.

ii) Collision resistance: no other probabilistic polynomial-time algorithm A, is exists, on input HK, that can generate two pairs (m_1, s_1) and (m_2, s_2) such that $m_1 \neq m_2$ and $h_{HK}(m_1, s_1) = h_{HK}(m_2, s_2)$ with non-negligible probability.

iii) **Trapdoor collision**: Given pairs (HK, TK), (m_1, s_1) and an additional message m_2, there exists a probabilistic polynomial-time algorithm that generates s_2 such that

– $h_{HK}(m_1, s_1) = h_{HK}(m_2, s_2)$.

– In s,if s_1 is uniformly distributed, then the s_2 distribution is computationally indistinguishable from uniform distribution in s.

B. Factoring-Based Trapdoor Hash Function

Selecting a random of two safe primes p and q (primes such as $p'= (p - 1)/2$ and $q' = (q - 1)/2$ are considered as primes) and compute n = pq. Select random an element g of order λ(n), where $\lambda (n) = lcm(p - 1, q - 1) = 2p'q'$. The public hash key HK is (n, g) and the private trapdoor key TK is (p, q). The trapdoor hash function $h_{HK}(m, s)$ is described in the following way as follows:

$$h_{HK}(m, s) = g^{m||s} \ mod\ n, \tag{4}$$

where || denotes concatenation. To show $h_{HK}(m,s)$ is a trapdoor hash function lying under the factoring assumption, one needs to show that it should satisfy the three main properties of a trapdoor hash function. The proof which shows $h_{HK}(m,s)$ as a factoring based trapdoor hash function can be found in [30].

For given pairs (HK,TK), (m_1, s_1) and an additional message m_2, to compute a trapdoor collision, we need to compute an s_2 such that $h_{HK}(m_1, s_1) = h_{HK}(m_2, s_2)$. According to equation (4), equivalently, we have $g^{m1||s1} = g^{m2||s2} \ mod \ n$. That is, we need to find an s_2 such that $2^k m_1 + s_1 = 2^k m_2 + s_2 \ mod \ \lambda \ (n)$, where k is the size of the auxiliary parameter s. Considered trapdoor key TK = (p,q), λ (n) can be computed in polynomial time and hence s_2 can be computed in polynomial time by solving the linear equation

$$s_2 = 2^k(m1 - m2) + s_1 \bmod \lambda (n).$$

C. Signature Scheme

In [30], paradigm in which hash-sign-switch, which combination of digital signature scheme and trapdoor hash family in (I,H) will be changed into an on-line / off-line signature scheme. Mainly, in the off-line phase, hash value generated by a signer whose value is committed to an selected message which is arbitrarily. In on-line phase, for the given message, signer will finds a collision of the trapdoor hash to the previously calculated hash value. The signature generated in the off-line phase and the collision point can be presented as the message generated signature in the on-line phase.

Suppose $h_{HK}(m,s)$ be a trapdoor hash function, here HK be the hash key, TK be the associated trapdoor key, VK be the verification key and SK be the signing key for any regular digital signature scheme. The given below describes the on-line / off-line signature scheme:

A) Key generation algorithm GEN: It generates a pair (SK,VK) using public-key generation algorithm and a pair (HK,TK) using the algorithm I. (SK,HK,TK) is the signing key and (VK,HK) is the verification key.

B) Signing algorithm SIGN: Here the signing key (SK, HK, TK) the signing algorithm operates as follows:

i) Off-line phase: The signer will randomly selects (m,s) and will computes $h_{HK}(m,s)$, then uses his secret key SK to sign $h_{HK}(m,s)$ and obtain $(S_{SK}(h_{HK}(m,s))$ and makes optionally $h_{HK}(m,s)$ during on- line phase to avoid re-computation.

ii) Online-phase: the message m′, the signer will finds collision of trapdoor hash for (m,s) such that $h_{HK}(m',s')=h_{HK}(m,s)$. the message signature m′ is defined as $(S_{SK}(h_{HK}(m,s)),s',h_{HK}(m,s))$.

C) Verification Algorithm VERF: Firstly, verify $(S_{SK}(h_{HK}(m,s)))$ using VK and $h_{HK}(m,s,)$ and computes $h_{HK}(m',s')$ to verify if $h_{HK}(m,s)=h_{HK}(m',s')$.

D. Generalized Diffie – Hellman Assumption (GDHA)

Let us consider A and B have their private keys x_A and x_B, and their corresponding public keys $y_A=g^{xB}modn$, respectively. Consider n=pq, where p and q are two large primes. Then it is assumed that only A and B can compute a shared secret $K_{A,B}=y^xA^B=y^xB^A \ mod \ n$. GDHA refers to the assumption that it is computationally infeasible to determine $K_{A,B}$ without knowing the private key x_A or x_B. It has been shown in [25] that GDHA is a valid assumption as long as factoring Blum-integers is hard.

E. User Authentication and Key Establishment Protocol

1) Registration at CA: Let certificate owner be A and verifier be the B. A to be register at CA and obtains GDC. The Certificate Authority CA generates an on-line / off-line digital signature, $(S_{SK}(h_{HK}(m',s'))$, for user A's statement m_A. Every owner have to keep the signature s_A in order to maintain the secret from verifier in authentication protocol. For providing secret component knowledge to the verifier, the owner conceals , during authentication phase following conceals the secret component to the verifier at GDHA. Our key establishment protocol and the user authentication is illustrated in the figure.

2) Protocol: The key establishment protocol and the authentication contain the following four steps:

a) The A user passes his information m_A and parameters $(S_{SK}(h_{HK}(m',s')),S_A,h_{HK}(m',s'))$, for the verifier B, where $S_A=g^{sA}mod \ n$.

b) After getting m_A and $(S_{SK}(h_{HK}(m',s'))$ is the signature of h(m′,s′) using the VK. Then computes $h_{HK}(m_A,S_A)=g^{2kmA}S_A mod \ n$, and verify if $h_{HK}(m_A,S_A)=h_{HK}(m',s')$, here k is secret exponent s_A length. If quality maintains true, then the verifier B first selects an integer v_B €[1,n-1], then computes $c_B = g^{vB}mod \ n$ and sends c_B to the user A. Otherwise protocol is stopped and the user authentication is failed.

c) The A user uses his secret s_A to compute Diffie-Hellman secret key $K_{A,B}=c^AB^Amod \ n$, $k'_{A,B} =D(K_{A,B})$. Then user A randomly selects an integer v_A€[n-1], computes $c_A=g^{vA}mod \ n$ and the response $Ack = h(K'_{A,B},c_B||c_A)$ represents a one-way keyed-hash function under the key $K'_{A,B}$. The user A sends Ack and c_A back to B.

d) After receiving the Ack and c_A from the user A, the verifier B uses his secret v_B to compute the Diffie-Hellman shared secret key $K_{B,A}=s^vA_B \ mod \ n$, $K'_{B,A}=D(K_{B,A})$, and checks whether $h(K'_{A,B},c_B||c_A)=Ack$ is true. If this verification is successful, the certificate owner A is authenticated by the verifier B and a one-time secret session key $c^vB_A=g^{vAvB}=c^vA_B \ mod \ n$ is shared between A and B. This key can provide perfect forward security.

In order to make verifier successfully authentication, in our protocol, the certificate owner needs to compute and sends valid parameters $(S_{SK}(h_{HK}(m',s')),S_A,h_{HK}(m',s'))$ and Ack to the verifier in steps 1 and 3. The parameters SA needs to satisfy.

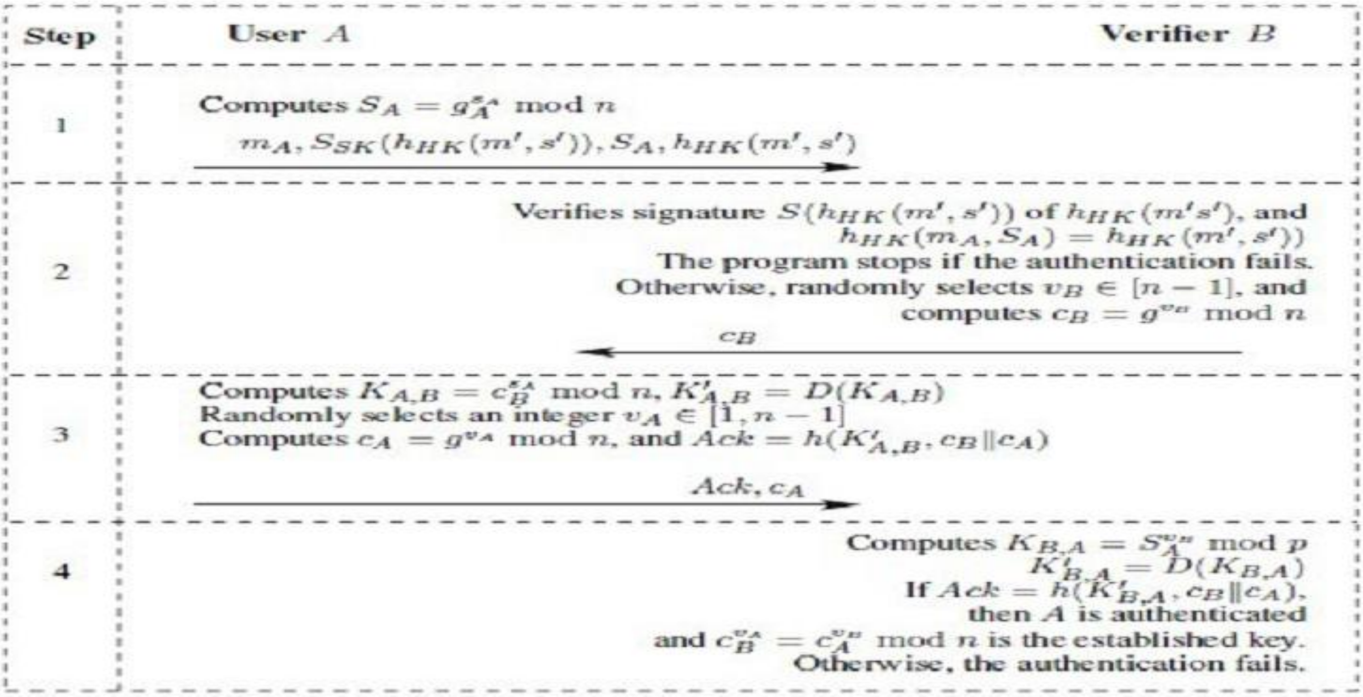
$$h_{HK}(m',s')=g^{2kmA}S_A \ mod \ n.$$

This parameter can be easily solved by anyone or is publicly available. However, we want to show that only the certificate owner A who knows the secret exponent of S_A can compute a valid Ack. This is because the verifier B can compute the one-time secret key $K_{B,A}$ used in generating Ack as $K_{B,A}=s^vA^B=g^{sAvB}mod \ n$. According to the GDHA, the certificate owner A who knows the secret exponent of S_A can also compute $K_{A,B}$ as $K_{A,B}=c^sB^A=g^{sAvB}=K_{B,A} \ mod \ n$. Thus, the certificate owner can interact with the verifier and be authenticated successfully.

Remark 4: In our proposed protocol, CA generates an on-line / off-line digital signature for each registered user. The CA does not actually need the trapdoor property of the one-way hash function. In fact, the CA does not need the trapdoor key. *It only needs to use the one-way hash property to compute a hash value SA*. Also, in order to construct an IF-based protocol, the CA needs to use the RSA signature to digitally sign the hash value h(m′,s′).

F. Security Analysis and Discussion

The security of this protocol relies on the combination of the security of the RSA signature, collision resistance of the one-way hash function and the GDHA. The On-line / Off-line digital signature is secure against scheme is secure against generic chosen-message attacks [30]. It has also proved that the trapdoor hash function is collision resistance [30]. Similar to the security analysis presented in Section III-E for the DL-based protocol, the proposed IF-based protocol also satisfies the properties of unforgeability, one-wayness and nontransfer-ability. The protocol also provides deniable authentication and protects privacy of the digital certificate.



Step	User A	Verifier B
1	Computes $S_A = g_A^{s_A} \bmod n$ $m_A, S_{SK}(h_{HK}(m', s')), S_A, h_{HK}(m', s')$ $\longrightarrow$	
2	$\longleftarrow$ c_B	Verifies signature $S(h_{HK}(m', s'))$ of $h_{HK}(m's')$, and $h_{HK}(m_A, S_A) = h_{HK}(m', s'))$ The program stops if the authentication fails. Otherwise, randomly selects $v_B \in [n-1]$, and computes $c_B = g^{v_B} \bmod n$
3	Computes $K_{A,B} = c_B^{s_A} \bmod n$, $K'_{A,B} = D(K_{A,B})$ Randomly selects an integer $v_A \in [1, n-1]$ Computes $c_A = g^{v_A} \bmod n$, and $Ack = h(K'_{A,B}, c_B \| c_A)$ Ack, c_A $\longrightarrow$	
4		Computes $K_{B,A} = S_A^{v_B} \bmod p$ $K'_{B,A} = D(K_{B,A})$ If $Ack = h(K'_{B,A}, c_B \| c_A)$, then A is authenticated and $c_B^{v_A} = c_A^{v_B} \bmod n$ is the established key. Otherwise, the authentication fails.

V. CONCLUSION

In this paper, we have proposed a novel design in using a SDC for user authentication and key establishment. In our design, user's public key will not be contained with SDC. As the user does not have any private and public key pair, this type of digital certificate is much easier than the X.509 public-key digital certificates. Our approach can be applied to both DL-base and IF-base public-key cryptosystems.

REFERENCES

[1] Generalized Digital Certificate for User Authentication and Key Establishment for Secure Communication, Lein Harn and Jian Ren, Senior Member IEEE 2372 – 2379 , *EEE Trans. Wireless Commun,* vol.10, No.7, July 2011

[2] Network Working Group, "Internet X.509 public key infrastructure certificate and crl profile, RFC: 2459," Jan. 1999.

[3] C. Tang and D. Wu, "An efficient mobile authentication scheme for wireless networks," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 1408- 1416, Apr. 2008.

[4] G. Yang, Q. Huang, D. Wong, and X. Deng, "An efficient mobile authentication scheme for wireless networks," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 168-174, Jan. 2010.

[5] J. Chun, J. Hwang, and D. Lee, "A note on leakage-resilient authenticated key exchange," *IEEE Trans. Wireless Commun.*, vol. 8, pp. 2274- 2279, May 2009.

[6] D. Chaum and H. van Antwerpen, "Undeniable signatures," *Advances in Cryptology - Crypto'89*, Lecture Notes in Computer Science, vol. 435, pp. 212-217, 1989.

[7] M. Bohøj and M. Kjeldsen, "Cryptography report: undeniable signature schemes," Tech. Rep., Dec. 15, 2006.

[8] X. Huang, Y. Mu, W. Susilo, and W. Wu, "Provably secure pairing-based convertible undeniable signature with short signature length," *Pairing- Based Cryptography -C Pairing 2007*, vol. 4575/2007 of *Lecture Notes in Computer Science*, pp. 367-391, Springer Berlin / Heidelberg, 2007.

[9] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated verifier proofs and their applications," *Advances in Cryptology – EUROCRYPT*, pp. 143-154, 1996. LNCS Vol 1070.

[10] D. Chaum, "Private signature and proof systems," 1996.

[11] R. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," *Advances in Cryptology-ASIACRYPT*, Lecture Notes in Computer Science, vol.
2248/2001, Springer Berlin / Heidelberg, 2001.

[12] J. Ren and L. Harn, "Generalized ring signatures," *IEEE Trans. Dependable Secure Comput.*, vol. 5, no. 4, Oct.-Dec., pp. 155-163, 2008.

[13] S. Saeednia, S. Kremer, and O. Markowitch, "An efficient strong designated verifier signature scheme," *ICISC 2003*, vol. 2836 of *Springer Lecture Notes in Computer Science*, pp. 40-54, 2003.

[14] C. Schnorr, "Efficient signature generation by smart cards," *J. Cryptology*, vol. 4, no. 3, pp. 161-174, 1991.

[15] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption << cost (signature) + cost (encryption)," *Advances in Cryptology - Crypto'97*, Lecture Notes in Computer Science vol. 1294, pp. 165-179, 1997. [16] F. Laguillaumie and D. Vergnaud, "Designated verifier signatures: anonymity and efficient construction from any bilinear map." IACR eprint. [17] R. Steinfeld, L. Bull, H. Wang, and J. Pieprzyk, "Universal designated verifier signatures," in *Asiacrypt'03*, vol. LNCS 2894, pp. 523-542, 2003.

[18] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. Assoc. Comp. Mach.*, vol. 21, no. 2, pp. 120-126, 1978.

[19] R. Steinfeld, H. Wang, and J. Pieprzyk, "Efficient extension of standard schnorr/rsa signatures into universal designated-verifier signatures," *PKC'04*, vol. Springer Lecture Notes in Computer Science of *2947*, pp. 86-100, 2004. [20] H. Lipmaa, G. Wang, and F. Bao, "Designated verifier signature
schemes: Attacks, new security notions and a new construction," in *ICALP'05*, 2005.

[21] Y. Li, W. Susilo, Y. Mu, and D. Pei, "Designated verifier signature: Definition, framework and new constructions," *Ubiquitous Intelligence and Computing*, vol. 4611/2007, Springer Berlin / Heidelberg, 2007.

[22] A. Mihara and K. Tanaka, "Universal designated-verifier signature with aggregation," in *Proc. Third International Conf. Inf. Technol. Appl.*, 2005. [23] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology: Proc. Crypto'84*, Lecture Notes in Computer Science vol. 196, (Berlin), pp. 47-53, Springer-Verlag, 1985.

[24] W. Diffle and M. E. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, pp. 644-654, 1976

[25] E. Biham, D. Boneh, and O. Reingold, "Breaking generalized Diffie-Hellman modulo a composite is no easier than factoring," *Inf. Process. Lett.*, vol. 70, pp. 83-87, 1999.

[26] T. A. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. 31, no. 4, pp. 469-472, 1985.

[3] Y Ramesh Kumar Completed his Master of Technology Degree from Andhra University in 2008. He is working as Associate Professor and Head of the Department for Computer Science and Engineering Department in Avanthi Institute of Engineering and Technology, An NBA accredited college, affiliated to Jawaharlal Nehru Technological University Kakinada. He is having 10 years of experience in Engineering Teaching Field. He is subject expertise in Programming Languages such as C, C++, Java, Web Technologies, Data Structures, PHP e.t.c. He had published books on C, Data Structures, Java and Web Technologies. His mostly research involves in application development scenario. He had guided more than 40 students i.e. of B.Tech and M.Tech in their academic research work. He was frequently invited as guest lecturer for the programming languages in several engineering colleges.

[2] Vasupalli Mahesh completed his Bachelor of Technology Degree and also Master of Technology Degree in Computer Science and Engineering Department from Jawaharlal Nehru Technological University Kakinada in 2010. He is working as an Associate Professor in Avanthi Institute of Engineering and Technology. He is having nearly 7 years of experience in engineering teaching field. He is subject expertise in Computer Network related concepts like Cryptography, Secure Communications, Network Security, Cloud Computing and Computer Networks. His more research work also related to these fields and guided more than 20 students in their research work i.e. of B.Tech and M.Tech in networking related concepts.

[1] Dasari Reventh Raj received his Bachelor of Technology Degree from Jawaharlal Nehru Technological University in 2011 and at present studying Master of Technology in Computer Science and Engineering Department from Avanthi Institute of Engineering Technology, which is affiliated to Jawaharlal Nehru Technological University.