Towards a Hybrid Public Key Infrastructure (PKI): A Review

Priyadarshi Singh, Abdul Basit, N Chaitanya Kumar, and V. Ch. Venkaiah

School of Computer and Information Sciences, University of Hyderabad,
Hyderabad-500046, India

Abstract. Traditional Certificate- based public key infrastructure (PKI) suffers from the problem of certificate overhead like its storage, verification, revocation etc. To overcome these problems, idea of certificate less identity-based public key cryptography (ID-PKC) was proposed by Shamir. This is suitable for closed trusted group only. Also, this concept has some inherent problems like key escrow problem, secure key channel problem, identity management overhead etc. Later on, there had been several works which tried to combine both the cryptographic techniques such that the resulting hybrid PKI framework is built upon the best features of both the cryptographic techniques. It had been shown that this approach solves many problems associated with an individual cryptosystem. In this paper, we have reviewed and compared such hybrid schemes which tried to combine both the certificate based PKC and ID-based PKC. Also, the summary of the comparison, based on various features, is presented in a table.

Keywords: Certificate-based PKI; Identity-based public key cryptography (ID-PKC); Hybrid PKI

1 INTRODUCTION

Public key infrastructure (PKI) and public key cryptography (PKC) [12] plays a vital role with four major components of digital security: authentication, integrity, confidentiality and non-repudiation. Infact, PKI enables the use of PKC through key management. The ”efficient and secure management of the key pairs during their whole life cycle” is the purpose of PKI, which involves key generation, key distribution, key renewal, key revocation etc [11]. One of the main tasks of PKI is to make the public key of an user authentic and valid. If the authenticity of public key is not guaranteed, the adversary may forge its use leading to the problem of repudiation, may decrypt messages that were encrypted for true user or may sign the documents in the name of true user. The Internet Engineering Task Force(IETF) Public Key Infrastructure X.509 (PKIX) [21] working group has been the driving force behind setting up a formal (and generic) model based on X.509 that is suitable for deploying a certificate-based architecture on the Internet. However, certificate-based approach is expensive and inefficient for any system that employs it.

1.1 Security of Public keys

Security of public keys implies the authentication of public keys which is necessary to avoid any frauds resulting from its misuse by an adversary. PKI provides a solution for it.

1.2 Overview of Certificate-based PKI and IBC

In traditional certificate-based PKI, user's public key generated by an user is authenticated with a digital certificate [22] issued by a trusted certification authority (CA). A digital certificate binds the public key of user with his/her identity and provides a means of "explicit authentication" to the user's public key in the sense that the authenticity of the public key is convinced to anyone by verifying the certificate. Any participant who wants to use other's public key must first verify the corresponding certificate to check the authenticity of the public key. Thus user's have to retrieve, verify, store, and manage others certificates that they are communicating with, which requires huge amount of storage, communication and computing to store, verify, and revoke certificates.

In 1984, Shamir [36] proposed the idea of Identity-based public key cryptography (ID-PKC) as an alternative to certificate based PKI. In ID-PKC, any publicly-known string (e.g. someone's email address) could be used as a public key and the corresponding private key is delivered to the proper owner of this string (e.g. the recipient of the email address) by a trusted key generator centre, KGC . This key generator must verify the user's identity before delivering the private key, of course, though this verification is essentially the same as that required for issuing a certificate in a typical PKI. Thus, an Identity-Based Encryption Scheme enables the deployment of a public-key cryptosystem without the prior setup of a PKI. The advantage of ID-PKC over certificate-based PKI is that distribution of public key is not required and thus the end users don't have to rely upon these public key certificates.

1.3 Problems associated with PKI and ID-PKC

Certificate-based PKI suffers from two main problems,namely scalability and certificate management[1]. Scalability indicates the extension of the PKI model in terms of its stakeholder whereas certificate management involves certificate generation, distribution, verification and revocation. It is not easy to scale certificate-based PKI as it requires to manage the trust relationship between intermediary CAs in hierarchy of trust, trust model usage etc. One major drawback of certificate-based PKI is the complicated and time-consuming verification process of certificates that requires to know and verify the certification path. To cope with these problems, ID-PKC was evolved but it couldn't offer true non-repudiation due to an inherent problem of Key escrow [29], i.e., KGC knows user's private key. Therefore, malicious KGC can decrypt cipher texts of the user and forge signatures with the name of the user. It also requires a secure channel between users and KGC to deliver private keys securely. Therefore, providing an escrow-free private key issuing mechanism [26] is an important issue to make the ID-PKC more practical in the real world. ID-PKC still requires certification of user's public key, thus complete eradication of certificates is still a question for researchers. Because of these inherent problems, ID-PKC is considered to be suitable for communications inside a small organization where KGC is fully trusted. Another problem associated with ID-PKC is that it lacks scalability in the sense that it's not easy to construct and manage the hierarchy of trust [16]. Other problems associated with it includes lack of support for a fine-grained revocation of identity [4], identity management overhead etc.

1.4 A Hybrid approach combining Certificate- based PKI and ID-PKC

In order to deal with the problems associated with certificate-based PKI and ID-PKC, a hybrid approach combining the best features of both the mechanisms had evolved. Here it is necessary to mention that some researchers use the term hybrid PKI for the combination of trust models of PKI[31] [13]. But we are not using in that sense although that will be considered while dealing with trust management aspect of PKI.

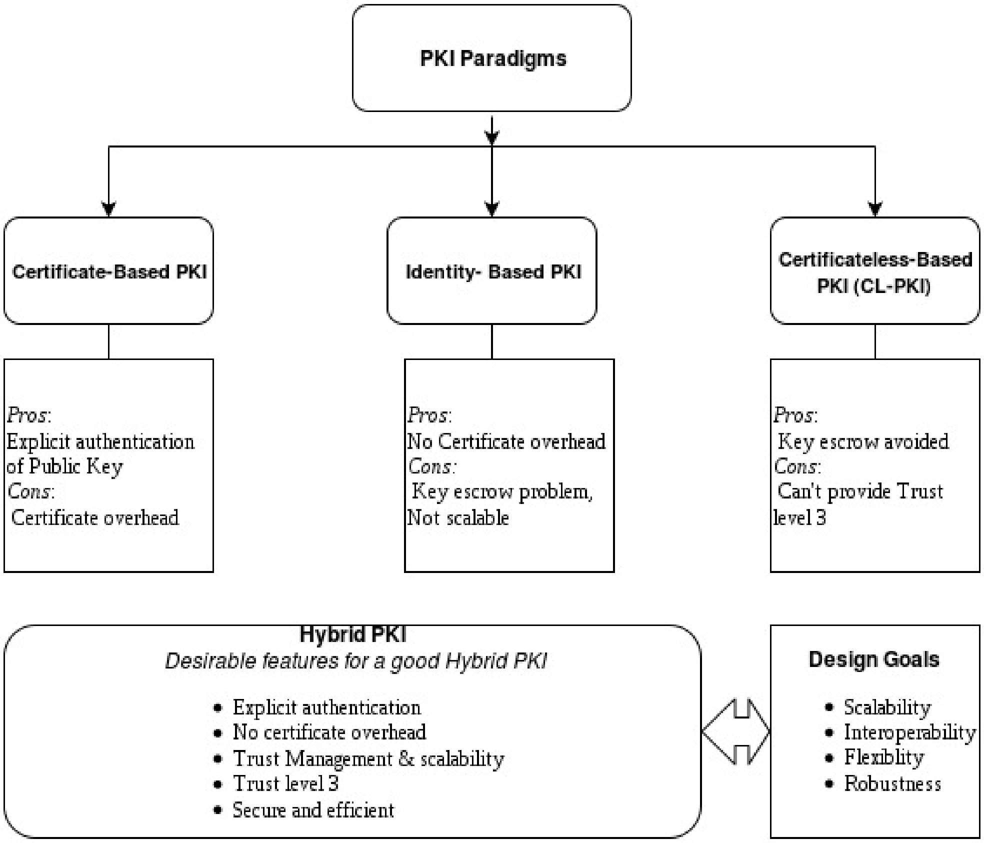
2 RELATED WORKS

There has been several works which try to combine certificate-based PKI and ID-PKC. Chen *et al*. [8] gives an idea to merge traditional PKI with identity-based encryption system and discussed various trust relationships among multiple authorities. They suggested to use CA-based PKI for creating upper level hierarchy of trust and ID-PKC for user level. This approach is both advantageous and scalable. Price *et al*. [34] considered the issue of interoperability between the two domains of traditional CA-based PKI and ID-based infrastructure, but they are silent on the implementation issues of the combined system.

An alternative to traditional certificate-based PKI, the concept of self certified keys was proposed by Girault [17] and further developed by Horster et. al. in [20]. But this scheme lacks explicit authentication of user's public key. Lee [27] solves this problem through self certificate's. In 2003, Al-Riyami and Paterson [1] introduced the concept of certificate less public key cryptography (CL-PKC), which addressed the key-escrow problem of the ID-PKC [5] and provided a lightweight infrastructure for public key certificate management. Gentry [15], independently, introduced the scheme of certificate-based encryption (CBE) which simplifies certificate management in traditional PKI systems by exploiting pairings. Lee [25] tried to provide an implementation of the idea of Chen *et al.* [8] and Price *et al.* [34] and further, presented the concept of Unified Public Key Infrastructure (UPKI) as a combined approach of supporting both certificate-based PKI and ID-PKC. Recently, Hassouna *et. al.*[19] proposed a hybrid PKI scheme based on Hassouna et. al. [18] that provides interoperability model between traditional certificate-based PKI and CL-PKI systems.

The rest of the paper is organized as follows. Section III makes us familiar with some background and preliminary information required for PKI schemes. In section IV, we have discussed various features and issues that need to be considered for a PKI scheme. Section V discusses various hybrid PKI schemes. A comparative analysis of these hybrid PKI schemes is made and presented in a table in section VI. Finally, conclusions are made in section VII with future work.

Fig. 1. Various PKI Paradigms


PKI Paradigms
Certificate-Based PKI
Identity- Based PKI
Certificateless-Based PKI (CL-PKI)
Pros: Explicit authentication of Public Key Cons: Certificate overhead
Pros: No Certificate overhead Cons: Key escrow problem, Not scalable
Pros: Key escrow avoided Cons: Can't provide Trust level 3
Hybrid PKI
Desirable features for a good Hybrid PKI
Explicit authentication
No certificate overhead
Trust Management & scalability
Trust level 3
Secure and efficient
Design Goals
Scalability
Interoperability
Flexiblity
Robustness

3 PRELIMINARIES

This section describes some of the preliminary information needed to describe PKI schemes.

3.1 Bilinear pairings

Let G_1 be an additive group of prime order q and G_2 be a multiplicative group of the same order. Let P denote a generator of G_1 and let the discrete logarithm problem (DLP) in these groups be believed to be hard. A bilinear pairing is a map e : $G1 \times G_1 \rightarrow G_2$ with the following properties:

1) *Bilinear:* $e(aQ_1, bQ_2) = e(Q_1, Q_2)^{ab}$, where $Q_1, Q_2 \in G_1$ and $a, b \in Z_q^*$
2) *Non-degenerate:* $e(P, P) \neq 1$ and therefore it is a generator of G_2.
3) *Computable:* There is an efficient algorithm to compute $e(Q_1, Q_2)$ for all $Q_1, Q_2 \in G_1$.

Typically, the map e will be derived from either the Weil or Tate pairing on an elliptic curve over a finite field. We refer to [5] and [2] for a more comprehensive description on how these groups, pairings and other parameters should be selected for efficiency and security. Now we describe some computational problems that form the basis of security for some PKI schemes.

3.2 Believed to be Hard Problems

- *Discrete Log Problem (DLP)*:
  Given two group elements P and Q in G_1, find an integer n such that $Q = nP$ whenever such an integer exists.
- *Computational Diffie-Hellman Problem (CDHP/DHP)*:
  For any a, $b \in Z_q^*$ given $\langle P, aP, bP \rangle$, compute abP.
- *Decisional Diffie-Hellman Problem (DDHP)*:
  Given P,aP,bP,cP, for any a,b,$c \in Z_q^*$, to decide whether $c \equiv ab\ (modq)$.
- *Bilinear Diffie-Hellman Problem (BDHP)*:
  For any a,b,$c \in Z_q^*$, given $\langle P, aP, bP, cP \rangle$,
  compute $e(P, P)^{abc} \in G_2$. An algorithm A has advantage ϵ in solving the BDHP in G_1, G_2, e if:
  $\Pr[A(P,\ aP,\ bP,\ cP) = e(P, P)^{abc}] = \epsilon$. Here the probability is measured over the random choices of a, b, $c \in Z_q^*$ and the random bits of A.
- *Gap Diffie-Hellman Problem (GDHP)*:
  A class of problems where DDHP is easy while CDHP is hard.

3.3 ID-based Cryptography

ID-based encryption (IBE) can be implemented using the Bilinear pairings and the GDHP. In 2001, Boneh and Franklin [5] implemented IBE using Bilinear pairings which relies on $BDHP$ assumption and the Random Oracle Model. The scheme is as follows:

- *Set up:* KGC specifies two groups G_1 and G_2, with a bilinear mapping $e : G_1 \times G_1 \rightarrow G_2$ and a generator P. It also specifies two hash functions.
  - $H_1 : \{0,1\}^* \rightarrow G_1$ (extract point from ID)
  - $H_2 : G_2 \rightarrow \{0,1\}^l$, where l is the length of a plaintext message (hash to the message space).

  KGC picks a master key $s_0 \in Z_q^*$ at random and computes his public key $P_0 = s_0 P$. KGC publishes parameters as $param = \{G_1, G_2, e, H_1, H_2, P, P_0\}$.
- *Extract:* Let Alice be a sender and Bob be a receiver. Bob requires a private key for his ID $\in \{0,1\}^*$ from KGC. For Bob's identity ID, the KGC computes Bob's public key as $Q_{ID} = H_1(ID)$ and the corresponding private key as $D_{ID} = s_0 Q_{ID}$ using short signature scheme [7] and sends D_{ID} to Bob through a secure channel. Bob can check the validity of his private key by $e(D_{ID}, P) = e(Q_{ID}, P_0)$
- *Encrypt*: Alice encrypts a message $m \in \{0,1\}^*$ with the public key of Bob. She computes Bob's public key as $Q_{ID} = H_1(ID)$. Then she picks a random number, r and computes $U = rP$ and $V = m \oplus H_2(e(Q_{ID}, P_0)^r)$ and sends the ciphertext $C = (U, V)$ to Bob.
- *Decrypt:* Bob decrypts the cipher text $C = (U, V)$ using his private key D_{ID} by $V \oplus H_2(e(D_{ID}, U) = m$. The decryption works because of the bilinear property of the map e,
  $e(D_{ID}, U) = e(s_0 Q_{ID}, rP) = e(Q_{ID}, P_0)^r$.

This scheme is not CCA2-secure, but can be made so with the Fujisaki-Okamoto construction [14], which assumes the Random Oracle Model.

4 Various PKI Comparison Factors and Design Goals

Various features which need to be considered while designing a PKI scheme are as follows:

4.1

PKI Comparison Factors

Nature of Scheme We can broadly classify a PKI scheme into one of the three categories i.e Certificate Based, Certificate less Based and Hybrid PKI scheme. Hybrid category consists of all those schemes which are intermediate or built up using the features of first two categories.

Trust Factor and Trust Levels In order to ensure the correct binding of a key-pair, there is a need for some trusted third party who helps in confirmation of authenticity of public key and assures that user possesses the correct corresponding private key. Now for the security aspect, it becomes necessary to question the trustworthiness of trusted party. Of course, absolute trust can't be made about it. So Girault [17] defined three levels of trust, based on the following trust assumptions made about the trusted third party (TTP).
Level 1: The TTP knows (or can easily compute) the private keys of user and therefore can impersonate any user at any time without being detected.
Level 2: The TTP does not know (or cannot easily compute) the private keys of user. But, the TTP can still impersonate a user by generating a false public key without being detected.
Level 3: The TTP does not know (or cannot easily compute) the private keys of user. Moreover, if TTP tried to impersonate an user by generating user's false public key, it will be detected.

A new level 0 is needed to be placed below level 1, which should consider that TTP should be trusted for correct binding of key-pair. The TTP should be trusted for correctly certifying the identity of user and generates the authenticated public key. In Certificate based PKI, since the authority is solely responsible for generating the users public key certificate, hence, the existence of two (or more) different certificates for the same user in the system is in itself a proof that the authority has cheated and thus detected. Thus, certificate-based PKI achieves trust level 3. On the other hand, in ID-PKC, TTP issues private keys to user and hence it achieves trust level 1.

Key Generation It includes generation of private key and the corresponding public key involving issues like who generates the key pair, when are the keys generated, where are the keys generated and how are the keys generated and the most important aspect of security i.e, the trust factor involved in these processes. Two things are required for generating a valid and authenticated key-pair. Firstly, public key should be authenticated i.e it should be linked to proper identity of user. Secondly, its corresponding private key must have an one to one relation in the sense that each key pair is uniquely identified in the system. There should be some assurance factor which affirms that user possess private key corresponding to the authenticated public key. Furthermore, the temporal issue between generation of private key and corresponding public key can have significant effects on the system [30]. In certificate-based PKI, the public key is generated at the same time as the private key while in ID-PKC, because of the separation between generation of private and public keys, a public key can be generated at a different time to the private key. Chen et. al.[9] use this feature of ID-PKC to enable the control of work-flow within a system.

Authenticity of Public Key Authenticity of Public key signifies that the particular public key belongs to the true user. This feature is needed to avoid any forgery or misuse of users public key. If the authenticity of user's public key is compromised, the adversary may forge the user in many ways, e.g., by replacing user's public key with his/ her own public key. The consequence of this mischief can be hazardous e.g decryption could not be possible as original user would not be able to decrypt with genuine private key and thereby missed the important messages that is to be recieved by him. Another problem lies with public key be is having the verification of signature. Here, a user will be unable to verify due to wrong public key he is having.

The authenticity of public key in a public key cryptosystem can be achieved in two ways: either explicitly or implicitly. During explicit authentication, the authenticity of the public key can be verified explicitly using the certificate issued by a certificate authority (CA), e.g., X.509 certificate [11]. In implicit authentication, it can be verified implicitly at the time when the key is used for encryption, signature verification, key exchange or any other cryptographic usage. The certification of public key in IBC is implicit, as a user can decrypt only if the TTP has issued a private key associated with the corresponding public key for a particular identity.

Guarantee Factor (G) This factor affirms the authenticity of public key of user. Depending on the kind of this guarantee [17], we may distinguish various types of PKI schemes. In certificate based PKI, the guarantee factor, G, is certificate provided by a trusted authority CA, whereas in ID-PKC, KGC generates the private key of user from his/ her public key corresponding for a particular identity, which itself is a guarantee for the authenticity of user's public key.

Non-Repudiation It ensures the prevention of an entity from denying having performed a particular action. In case, authenticity of public key is violated, then the adversary can repudiate a transaction; e.g., a signer can later deny that he/she signed a particular message to other party. So in order to ensure non-repudiation, it is required to maintain the authenticity of user's public key. In certificate-based PKI, this assurance comes from the use of digital signatures [22] of a trusted authority, CA, while in IB-PKC, true non-repudiation can't be achieved due to key escrow problem [38].

Assurance Factor and Proof of Possession (*PoP*) It ensures that the user having particular public key possess its corresponding private key. This is needed because it ensures non- repudiation in the sense that signer can't deny at some later time that he had signed this particular message. A more general term for it could be an "Assurance Factor", which assures in some general sense that a user possess a particular private key corresponding to its authenticated public key. This can also be termed as a Binding factor, which binds the key-pair. This binding factor should be made trusted otherwise the problem will remain the same. In certificate-based PKI, this assurance factor is achieved through PoP by certificate authority. In ID-PKC, the assurance factor is achieved through KGC itself, which binds the key-pair.

Key renewal Each key pair is valid for a certain time period, after which it must be renewed either by user or by some authority like CA as per the policies. Key pairs can also be generated either by some trusted authority like CA or by user itself. If the key pair is chosen by user, he can renew his key pair with or without the interaction of CA, while keeping the authenticity of CA's certification. In certificate based PKI, since key pair is generally chosen by user, he cannot renew it by himself without interacting with CA because corresponding certificate should also be renewed. In ID-PKC, key renewal is performed by KGC itself for a particular identity attributes registered.

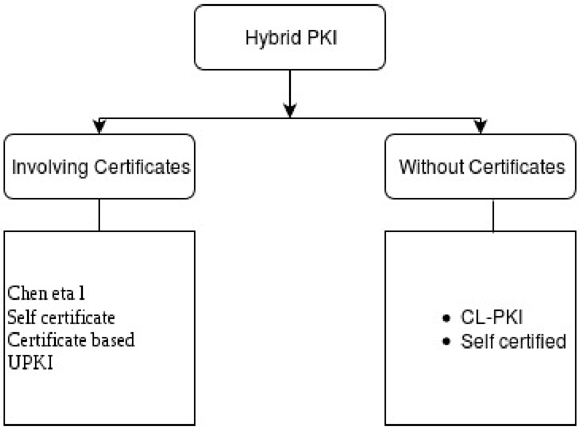
Key Revocation Every key is to be revoked before its expiration due to the reasons mentioned in Stallings [37]. In certificate-based PKI, key revocation is done through popular methods like certificate revocation list (CRL)[21], the Online Certificate Status Protocol (OCSP) [28]. The certificate's validation becomes too complicated and complex for reasons like to know the certification path, the revocation status of subsequent certificates in the "chain of certificates" etc. Key revocation is even more cumbersome in ID-PKC. Since an ID is a fixed information given for a person, like name. So it is hard to revoke ID. Further study is required for the ID revocation problem.

Computation & Communication This factor analyses the computation and communication parameters involved in the verification of authentication of public key; e.g., verification of CA's signature, verification of CA's certification, online communication required, Hash functions used etc. To verify the authenticity of public key, certificate-based scheme requires one verification of CA's signature.

Hierarchy of Trust Scalabilty in PKI deals with the hierarchy of trust. In order to scale any PKI system, it is necessary to consider architectural model of PKI [13]. Trust problem arises among the intermediary trusted authorities and user. If end user is not trusted with the public key of his immediate trusted authority, CA, he may wish to validate the "chain of certificate" or "chain of trust"[37]. Certificate-based PKI can be deployed to authenticate users in large scale using hierarchical groups, while IBC is generally used to authenticate users in a closed highly trusted group.

5 VARIOUS PKI SCHEMES

We have restricted our study only to those hybrid PKI schemes which require some kind of certification. The reason for this selection is attributed to the reason that CL-PKC hybrid PKI schemes suffer from the problem of scalability and hence they failed to acheive hierarchy of trust in PKI environment.



5.1 Chen et. al.'s Hybrid PKI / IBE System

Replacement of certificate-based PKI with IBE is not practically feasible due to the scalability issues. Chen et. al.[8] proposed the idea of hybrid PKI/IBE, in which he proposed to use both the paradigms in a single framework. He suggested to use certificate-based PKI to create upper level of hierarchy of trust and IBE to be used at user level. As such end users don't need to maintain other users certificates but he still needs to manange certificates of trusted authorities.

The scheme seems to have some advantage as it reduce the burden on users and it solves the problem of scalability but on the other hand it increases the complexity of the system. Further, the scheme still suffers from the problem of key escrow of IBE, trust problems of upper level hierarchy of certificate-based PKI.

5.2 Self Certified Key Scheme

This scheme was given by Girault [17] as an alternative to certificate based PKI and extended in many ways in [20]. The scheme can be considered as a certificate less hybrid scheme as it avoids the use of certificates while preserving the certificate-based PKI model. Actually, this scheme consists of a Self certified public key which itself contains the signature of the CA as certification information. This scheme was designed to satisfy a computationally unforgeable relationship between three attributes of a user: his identity I, his public key y and the corresponding private key x. A self-certified key scheme of [32] uses the Schnorr signature scheme [35] as underlying signature scheme. The schnorr signature scheme, which is proven to be secure under the random oracle model is as follows:

Schnorr signature scheme: A certification authority (CA) chooses large primes p, q with $q|p-1$, a generator g of a multiplicative subgroup of Z_p^* with order q and a collision resistant hash function, h. He publishes p, q, g and h. Assume that a signer Alice has a private key x_A and the corresponding public key $y_A = g^{x_A}$. To sign a message m, Alice chooses a random number $k \in_R Z_q^*$ and computes $r = g^k$, $s = x_A h(m,r) + k$. Then the triple (m, r, s) becomes the signed message. The verification of signature is checked by $g^s = y_A^{h(m,r)} r$. This signature scheme has been proven to be secure under the random oracle model [33].

We review the secure key issuing protocol of [20]. CA has a private key x_{CA} and the corresponding public key $y_{CA} = g^{x_{CA}}$. CA signs Alice's identity ID_A using a weak blind Schnorr signature [20]. Using the following interactive protocol, CA issues a self-certified key pair (x_A, y_A) to Alice.

Secure Key Issuing Protocol: The certification authority chooses $\tilde{k_A} \in_R Z_q^*$ as before and computes $\tilde{r}_A := g^{\tilde{k}_A}$ $(mod\ p)$. She transmits $\tilde{r}_A$ to Alice, who chooses a random $a \in_R Z_q^*$ and computes $r_A := \tilde{r}_A \ . \ g^a (mod\ p)$. Alice sends $(ID_A,\ r_A)$ to the CA, who computes the signature parameter $\tilde{s}_A := x_{CA}.h(ID_A, r_A) + \tilde{k}_A$. This value $\tilde{s}_A$ is transmitted to Alice who obtains her secret key $x_A := \tilde{s}_A + a$ $(mod\ q)$. The tuple $(r_A, x_A) := S(x_{CA}, ID_A)$ is a signature on her identity. Her corresponding public key is computed as $y_A := g^{x_A} \equiv y_{CA}^{h(ID_A, r_A)}.r_A$ $(mod\ p)$. Alice publishes (r_A, ID_A) and keeps x_A as her private key.

The secret key x_A is hidden to CA, as it is blinded by the random value a. Thus, the protocol reaches trust level 3. As only CA is capable of issuing valid self-certified keys, the existence of two different valid keys for the same user (e.g. in the case, when CA impersonates Alice) proves that the authority was cheating. Thus such fraud is detectable by users. One obvious advantage of this scheme is the reduction of storage and computation (they don't require hash functions at the authority level) while secret keys are still chosen by the user himself and remain unknown to the authority, avoiding the key escrow problem [38].

Authenticity of Public key can be verified implicitly through the subsequent use of the correct private key x_A and the guarantee comes from the the public key (i.e $G = y_A$) itself as it is self certified key. One may criticize this scheme as it lacks explicit authentication of public key y_A. Any adversary can generate a similar public key y'_A by modifying the public parameter r'_A and Alice's identity ID_A by $y'_A = y_{CA}^{h(ID_A, r'_A)} r'_A$, which can be distinguished only after any successful communication with the owner. CA may maintain a trusted public directory for r_A, but it requires an extra online communication effort as compared with the offline verifiability of the certificate-based schemes.

5.3 Self- Certificate Scheme

To overcome the problem of explicit authentication of self-certified key, Lee [27] proposed a hybrid scheme of Self-certificate. This is a user generated certificate for the self certified public key by signing the public key and relevant information with the private key corresponding to the public key. In this scenario, user can renew his key pairs by himself without any interaction with CA, while keeping the authenticity of CA's certification. CA can also use the same revocation mechanism as that of certificate-based scheme to revoke an issued key. The definition of self-certificate given by Lee [27] is:

Definition 1 (Self-Certificate) *Let (x_A,y_A) be the key pair issued by CA, r_A be Alice's public parameter, and ID_A be Alice's identity. Alice signs on ID_A, y_A with her private key x_A to generate*

$$SelfCert_A = Sig_A(ID_A, y_A, r_A)$$

Then $SelfCert_A$ is called self-certificate for the public key y_A.

The scheme is a slight modification of Girault's [17] self certified scheme. The only modification made by Lee [27] is to replace the identity (ID_A) of Alice by a self certificate signed by Alice herself on behalf of CA.

Generation of Self-Certificate: To generate self-certificate for a self-certified key, the modifications made in the secure key issuing protocol are as follows. CA chooses $\tilde{k}_A \in_R Z_q^*$, computes $\tilde{r}_A = g^{\tilde{k}_A}$, and transmits $\tilde{r}_A$ to Alice. Alice chooses $a \in_R Z_q^*$, computes $r_A = \tilde{r}_A g^a mod p$, sends (ID_A, r_A) to CA. CA prepares certification information CI_A depending on her policy (including PoP as part of registration). For example, she uses Alice's identity, CA's identity, Alice's public parameter r_A, certificate serial number, validity period, CA's public key and any other relevant extension information.

$$CI_A = [ID_A||ID_{CA}||r_A||CertNo||Period||y_{CA}||E_{xt}]$$

CA computes the signature parameter $\tilde{s}_A = x_{CA}h(CI_A) + \tilde{k}_A$ and sends (CI_A , $\tilde{s}_A$) to Alice. Alice obtains her private key $x_A = \tilde{s}_A + a$. The tuple (r_A, x_A) is CA's signature on certification information CI_A. She verifies the validity of CA's signature by

$$y_A \equiv g^{x_A} = y_{CA}^{h(CI_A)} r_A \tag{1}$$

Then her public key is y_A. Alice signs (CI_A, y_A) with her private key x_A to generate the self-certificate of y_A.

$$SelfCert_A = Sig_A(CI_A, y_A)$$

This signature can be considered as a proxy signature [23][20], delegated by CA.

Verification of self-certificate: When the $SelfCert_A$ is presented, anyone can explicitly verify the validity of y_A by Checking the validity of Alice's signature in

$SelfCert_A$ using y_A and the validity of CA's certification by checking equation (1). Thus, this scheme requires one verification of user's signature and one extra exponentiation for the verification of CA's certification.

The scheme enjoys the trust level 3 as the scheme is based on self certified key [17]. Authenticity of user's public key can be verified explicitly and the guarantee factor, $G = selfcertificate$, delegated by CA. One advantage of this scheme is that Alice can renew her key pairs without any interaction with CA, while keeping CA's certification relation.A major problem associated with this scheme is self-certificate overhead which is same as with certificate overhead in traditional certificate-based PKI scheme.

5.4 Certificate based encryption (CBE) Scheme

Gentry [15] independently introduced this scheme with a view to simplify revocation in traditional PKIs. It simplified certificate management in traditional PKI systems by exploiting pairings. This scheme combines the best aspects of identity-based encryption (implicit certification) and public key certification (no escrow) and can be placed as an intermediate hybrid scheme.

In Gentry's model, each client generates its own public key/secret key pair and requests a certificate from the CA. The CA uses an IBE scheme to generate the implicit certificate, which can be used explicitly as proof of current certification and as a decryption key. The implicit certification allows this scheme to eliminate third-party queries on certificate status [15]. Basically, a signer entity B's private key consists of two components: the first component is chosen by entity itself and keeps private and the second component which is time-dependent and is issued to B on a regular basis by a CA and can be made public. Corresponding to these private components, there are two public key components. The first of these is chosen by B while the second can be computed by other entity, A using some public parameters of the CA together with the current time value (i) and the assumed value of B's public key. The second private component acts as an $implicit\ certificate$ for relying parties: one that a relying party can be assured is only available to B provided that B's certification has been issued for the current time period by the CA. This approach provides an implicit revocation mechanism for PKI and thus simplify certificate revocation of traditional PKI system. There is no need for A to make any status checks on B's public key before encrypting a message for B. The scheme is free from Key escrow problem of IBC [36].

The following is a Basic CBE [15] scheme based on Boneh-Franklin [5] approach: Let a randomized algorithm IG be a Bilinear Diffie-Hellman (BDH) parameter generator which takes a security parameter $k>$ 0 and generates groups G_1, G_2 of some prime order q and an admissible pairing $e : G_1 \times G_1 \rightarrow G_2$. The CA picks an arbitrary generator $P \in G_1$ and a random secret $s_C \in Z_q^*$ and computes $Q = s_C P$. He also chooses two cryptographic hash functions $H_1 : \{0, 1\}^* \rightarrow G_1$ and $H_2 : G_2 \rightarrow$

$\{0,1\}^n$ for some n. The system parameters are $params = (G_1, G_2, e, P, Q, H_1, H_2)$. The message is $M = \{0,1\}^n$ and CA's secret is $s_C \in Z_q^*$. The CA uses its parameters and its secret to issue certificate to Bob as follows. Here we are assuming that Bob's secret key/ public key pair as $(s_B, s_B P)$, where $s_B P$ is computed according to the parameters issued by the CA.

Certification:

- Bob sends $Bobsinfo$ to the CA, which includes his public key $s_B P$ and any other necessary identifying information.
- After verifying Bob's information (including PoP), CA computes $P_B = H_1(s_C P, i, Bobsinfo) \in G_1$ in period i and then an implicit certificate $Cert_B = s_C P_B$ and send it to Bob. This certificate can be made public for having the advantage of explicit authentication as in Certificate-based PKI.

Bob also signs $Bobsinfo$, producing $s_B P'_B$. where $P'_B = H_1(Bobsinfo)$. Now the $S_{Bob} = s_C P_B + s_B P'_B$ is a two person aggregate signature, as defined in [6], which will act as his decryption key.

Encryption: To encrypt $m \in M$ using $Bobsinfo$, Alice computes two public components as, $P'_B = H_1(Bobsinfo) \in G_1$ for Bob's public key and $P_B = H_1(Q, i, Bobsinfo) \in G_1$ for implicit certificate. Then she chooses a random parameter $r \in Z_q^*$ and sets the ciphertext as :
$C = [rP, M \oplus H_2(g^r)]$ where $g = e(s_C P, P_B)e(s_B P, P'_B) \in G_2$.

Decryption: Bob decrypts the message $[U, V]$ only if he is in possession of both the private components, i.e., $S_{Bob} = s_C P_B + s_B P'_B$ as $M = V \oplus H_2(e(U, S_{Bob}))$ where $U = rP$ and $V = M \oplus H_2(g^r)$.

Since this scheme is based on Boneh and Franklin approach [5], it is not secure against adaptive chosen-ciphertext, but can be made so with Fujisaki-Okamoto transform [14].

This scheme provides the service of issuing a secure key successfully avoiding the key escrow problem. Moreover a secure channel is not required; an implicit certificate, $Cert_B$ can be sent over a public channel or published thereby losing the advantage of ID-PKC and the scheme is reduced to certificate-based PKI. CBE provides an implicit revocation mechanism for PKIs due to involvement of implicit certificate. Authentication of public key is implicit as there is no need for A to make any status checks on B's public key before encrypting a message for B; rather A's assurance that only B can decrypt comes through trusting the CA to properly update and distribute the second components of private keys and the guarantee factor $G = Implicit\ certificate$. This model do not require the use of explicit certificates as in certificate-based PKI and thereby scheme is more efficient as argued in [3]. The security of CBE depends critically on the CA binding the correct public key into B's implicit certificate in each time period. This approach simplifies revocation in PKIs: as there is no need for A to make any status checks on B's public key before encrypting a message for B. So there are no CRLs and no requirement for

OCSP. This scheme suffers from one major drawback: the CA needs to issue new implicit certificates to every user for every time period. A granularity of one hour per time period is suggested in [15]; this substantially adds to the computation and communication at the CA site with even a small user base. This scheme is silent on the issue of scalability and assumes only the single CA based hierarchical model of PKI architecture [13] and have not discussed the applicability of the scheme to increased level of CAs.

Table 1. Comparison of Various PKI Schemes

Features	Certificate based scheme	Self-certified key scheme	Self-certificate scheme	Identity-based scheme(ID-PKC)	Certificate based Scheme(CBE)	Unified Public Key Infrastructure (UPKI)
Type of scheme	Certificate based	Hybrid	Hybrid	Identity based	Hybrid	Hybrid
Authentication	Explicit	Implicit	Explicit	Implicit	Implicit	Explicit
Trust Level	3	3	3	1	3	3
Guarantee Factor (G)	Certificate	Public key	Self Certificate	Private key	Implicit Certificate	Certificate and Private Key
Non-repudiation	Y	N	Y	N	Y	Y
User Controlled Key Renewal	N	Y	Y	N	N	N
Key Revocation	Y	N	Y	Y	Y	Y
Assurance Factor	Y (PoP)	N	Y	Y (KGC)	Y	Y
Computation / Communication	1V	1E+1C	1V+1E (2E)	1C+1H	1E+1C+2H + (1V)	1V+1C+1H
Scalability	Y	N	N	N	N	Y

* *V: Signature verification, E: Exponentiation, c: Online communication, H: Hash function, Y: Yes, N: No

5.5 Unified Public Key Infrastructure (UPKI) Scheme

Lee [25] provided an implementation of the idea of hybrid PKI/ IBE of Chen *et al.* [8] and discussed the inter domain communication between entities belonging to this new UPKI framework as discussed by Price *et al.* [34]. He further presented the concept of UPKI in which both Certificate-based PKI and ID-PKC are provided to users in a single framework. He further claimed that as the end user's interact using

IBC, the scheme is not required to manage the certificates of other user's except for the certificates of trusted authorities, thus this scheme mitigates the certificate overhead for end user's. This scheme claims to add an extra effort of IBC with some efficient gain over traditional certificate-based PKI. This framework assumes the presence of a trusted authority, key generation and certification authority ($KGCA$) who plays the role of both KGC and CA. $KGCA$ checks the identity credentials of user and issues a certificate to the user for his/her public key X as part of CA and an ID-based partial private key to the user as part of KGC. Apart from $KGCA$, this scheme also assumes the requirement of multiple Key privacy agents ($KPA's$) like in [26] which provides the service of key privacy.

For implementing this hybrid scheme, Lee[25] proposed that user should have two pairs of keys. One for certificate-based PKI, i.e., certified key pair (x, X) with $Cert(ID, X)$ and second one for ID-PKC, i.e., ID-based key pair (SK_{id}, PK_{id}), where $PK_{id} = H(ID)$, so that the user can use both paradigms according to his/her need. The scheme is as follows:

- *Certificate Issuing*: As in traditional certificate-based PKI, user, upon registering (including PoP) to $KGCA$, will be issued a certificate $Cert(ID, X)$ for the public key X chosen by him/her using some standard like X.509 [22].
- *Partial Private key issuing*: On request of user, KGC issues a partial private key SK'_{ID} for ID after verifying the user's certificate and proof of possession of x.
- *Key Privacy Services*: User requests key privacy services from $n - KPAs$. Each KPA after verifying the user credentials, signs SK'_{ID} with its private key and sends it to user through a secure channel [26].

User after collecting valid t-signatures of KPAs retrieve his/her ID-based private key SK_{ID} using some (t, n)- threshold key issuing protocol [24]. Now user has two pairs of keys; certified key pair (x, X) and IBC key pair (SK_{ID}, PK_{ID})

UPKI achieves the trust level 3. This scheme provides some efficiency over CA-based PKI as this scheme do not require certificate at user level. Infact, UPKI provides implicit authentication at user level. But for upper level hierarchy of trust, scheme requires same explicit certification process for intermediary CA's.

One disadvantage of this scheme is high communication and computation overhead as it involves inefficiency of introducing multiple $KPA's$ and requirement of a secure channel. Secondly, use of Pairings makes it impracticable to use it in real PKI scenario, e.g., PKIX [10]. Another problem is with revocation which is not easy as it involves ID-based cryptography [36].

6 COMPARISON OF VARIOUS SCHEMES

In this section, we have compared various hybrid PKI schemes based on various PKI features and design issues. Table I summarizes these results. The table shows

UPKI scheme is better for a system to be scalable and in overall fulfillment of other features as par with other schemes.

7 CONCLUSIONS AND FUTURE WORK

We have reviewed the various hybrid PKI schemes involving certificates and a comparison is made based on various features. We believe some kind of certification is required in order to solve the problem of explicit authentication and scalability related issues. UPKI achieves all these problems except with some communication and complexity overhead. Further research is required to minimize these overhead. As a future work, a better and more efficient hybrid PKI scheme needs to be designed while considering various design goals of PKI.

8 Compliance with Ethical Standards

8.1 Disclosure of potential conflicts of interest

Conflict of Interest: The authors declare that they have no conflict of interest.

8.2 Research involving human participants and/or animals

This article does not contain any studies with human participants performed by any of the authors.

References

1. Sattam S Al-Riyami and Kenneth G Paterson. Certificateless public key cryptography. In *Asiacrypt*, volume 2894, pages 452–473. Springer, 2003.
2. PSLM Barreto, Hae Yong Kim, Ben Lynn, and Michael Scott. Efficient algorithms for pairing-based cryptosystems. In *Crypto*, volume 2, pages 354–368. Springer, 2002.
3. Alexandra Boldyreva, Marc Fischlin, Adriana Palacio, and Bogdan Warinschi. A closer look at pki: Security and efficiency. *Public Key Cryptography–PKC 2007*, pages 458–475, 2007.
4. Alexandra Boldyreva, Vipul Goyal, and Virendra Kumar. Identity-based encryption with efficient revocation. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 417–426. ACM, 2008.
5. Dan Boneh and Matt Franklin. Identity-based encryption from the weil pairing. In *Advances in CryptologyCRYPTO 2001*, pages 213–229. Springer, 2001.
6. Dan Boneh, Craig Gentry, Ben Lynn, and Hovav Shacham. Aggregate and verifiably encrypted signatures from bilinear maps. In *Eurocrypt*, volume 2656, pages 416–432. Springer, 2003.
7. Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the weil pairing. *Advances in CryptologyASIACRYPT 2001*, pages 514–532, 2001.
8. Liqun Chen, Keith Harrison, Andrew Moss, David Soldera, and Nigel P Smart. Certification of public keys within an identity based system. In *International Conference on Information Security*, pages 322–333. Springer, 2002.

9. Liqun Chen, Keith Harrison, David Soldera, and Nigel P Smart. Applications of multiple trust authorities in pairing based cryptosystems. In *Infrastructure Security*, pages 260–275. Springer, 2002.
10. Santosh Chokhani, Warwick Ford, Randy Sabett, Charles Merrill, and Stephen Wu. Internet x. 509 public key infrastructure certificate policy and certification practices framework. Technical report, 2003.
11. Dave Cooper. Internet x. 509 public key infrastructure certificate and certificate revocation list (crl) profile. 2008.
12. Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE transactions on Information Theory*, 22(6):644–654, 1976.
13. Zakia El Uahhabi and Hanan El Bakkali. A comparative study of pki trust models. In *Next Generation Networks and Services (NGNS), 2014 Fifth International Conference on*, pages 255–261. IEEE, 2014.
14. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In *Crypto*, volume 99, pages 537–554. Springer, 1999.
15. Craig Gentry. Certificate-based encryption and the certificate revocation problem. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 272–293. Springer, 2003.
16. Craig Gentry and Alice Silverberg. Hierarchical id-based cryptography. *Advances in cryptologyASIACRYPT 2002*, pages 149–155, 2002.
17. Marc Girault. Self-certified public keys. In *Advances in CryptologyEUROCRYPT91*, pages 490–497. Springer, 1991.
18. Mohammed Hassouna, Bazara Barri, Nashwa Mohamed, and Eihab Bashier. An integrated public key infrastructure model based on certificateless cryptography. *International Journal of Computer Science and Information Security*, 11(11):1, 2013.
19. Mohammed Hassouna, Bazara IA Barry, and Eihab Bashier. A new level 3 trust hierarchal certificateless public key cryptography scheme in the random oracle model. *IJ Network Security*, 19(4):551–558, 2017.
20. Patrick Horster, Markus Michels, and Holger Petersen. Hidden signature schemes based on the discrete logarithm problem and related concepts. In *Communications and Multimedia Security*, pages 160–177. Springer, 1995.
21. Russell Housley, Warwick Ford, William Polk, and David Solo. Internet x. 509 public key infrastructure certificate and crl profile. Technical report, 1998.
22. Recommendation X ITU-T. 509 the directory-authentication framework. *International Telecommunication Union, Geneva, Switzerland*, 1993.
23. Seungjoo Kim. *Improved privacy and authenticity in digital signature/key management.* PhD thesis, Ph. D. Thesis, SungKyunKwan University, 1998.
24. K Phani Kumar, G Shailaja, and Ashutosh Saxena. Secure and efficient threshold key issuing protocol for id-based cryptosystems. *IACR Cryptology ePrint Archive*, 2006:245, 2006.
25. Byoungcheon Lee. Unified public key infrastructure supporting both certificate-based and id-based cryptography. In *Availability, Reliability, and Security, 2010. ARES'10 International Conference on*, pages 54–61. IEEE, 2010.
26. Byoungcheon Lee, E Dawson, and S Moon. Efficient and robust secure key issuing protocol in id-based cryptography. In *Preproceedings of the 6-th International Workshop on Information Security Applications (WISA 2005)*, pages 267–280.
27. Byoungcheon Lee and Kwangjo Kim. Self-certificate: Pki using self-certified key. In *Proc. of Conference on Information Security and Cryptology*, volume 10, pages 65–73, 2000.
28. Michael Myers, Rich Ankney, Ambarish Malpani, Slava Galperin, and Carlisle Adams. X. 509 internet public key infrastructure online certificate status protocol-ocsp. Technical report, 1999.
29. JoongHyo Oh, KyungKeun Lee, and SangJae Moon. How to solve key escrow and identity revocation in identity-based encryption schemes. In *ICISS*, pages 290–303. Springer, 2005.

30. Kenneth G Paterson and Geraint Price. A comparison between traditional public key infrastructures and identity-based cryptography. *Information Security Technical Report*, 8(3):57–72, 2003.
31. Radia Perlman. An overview of pki trust models. *IEEE network*, 13(6):38–43, 1999.
32. Holger Petersen and Patrick Horster. Self-certified keys-concepts and applications. In *Proc. Communications and Multimedia Security*, volume 97, pages 102–116. Springer, 1997.
33. David Pointcheval and Jacques Stern. Security proofs for signature schemes. In *Eurocrypt*, volume 96, pages 387–398. Springer, 1996.
34. Geraint Price and Chris J Mitchell. Interoperation between a conventional pki and an id-based infrastructure. *Lecture notes in computer science*, 3545:73, 2005.
35. Claus-Peter Schnorr. Efficient identification and signatures for smart cards. In *Conference on the Theory and Application of Cryptology*, pages 239–252. Springer, 1989.
36. Adi Shamir et al. Identity-based cryptosystems and signature schemes. In *Crypto*, volume 84, pages 47–53. Springer, 1984.
37. William Stallings. *Cryptography and network security: principles and practices*. Pearson Education India, 2006.
38. Tsz Hon Yuen, Willy Susilo, and Yi Mu. How to construct identity-based signatures without the key escrow problem. *International Journal of Information Security*, 9(4):297–311, 2010.