Use Cases of Implicit Authentication and Key Establishment with Sender and Receiver ID Binding

Dan Forsberg
Nokia Research Center
dan.forsberg@nokia.com

Abstract

Common public key based authentication and key establishment (AKE) mechanisms with related Public Key Infrastructures (PKI) are relatively heavy. In this paper we explore public key and identity (ID) based AKE protocols, which do not require certificate authorities. We apply this approach to the symmetric key cryptography by creating a novel and lightweight symmetric key AKE protocol based on sender and/or receiver IDs. To show, how our protocol can be used in places, where common PKI does not perform well, we present four new and interesting use cases. One of them is a simple and efficient ISP service user AKE with the help of telecom operator's Short Message Service (SMS). Our protocol can also be used for wireless network access AKE.

1. Introduction

Common public key based A*uthentication and Key Establishment* (AKE) protocols require Public Key Infrastructures (PKI) so that the end points can be authenticated based on a valid certificate signed by a common *Certificate Authority* (CA). In practice, for example in many Internet based services, only the web servers are required to have a valid certificate and client authentication is handled based on a local username and password database. One could ask that why aren't these two (PKI and local username and password databases) combined together to make the system easier to manage and deploy.

- In this paper we explore *Identity Based Cryptography* (IBC) AKE protocols.
- We apply this abstraction level to the symmetric key cryptography by creating a novel and lightweight shared secret based AKE protocol based on sender and/or receiver ID binding.
- We describe new and interesting use cases with our protocol to show how it could be used in places where common public key based AKE protocols do not perform well.

We provide background to IBC in general and especially IBC AKE (Section 2). We explain in high level the basic theorems behind and provide references to sources with proper proofs and deeper overviews. We also give some examples of the existing applications with IBC AKE protocols and identity binding schemes in symmetric key cryptography. Then, we take the same analogy to the symmetric key world and describe our new and simple symmetric key based identity binding AKE protocol (Section 3). After that we provide four example use cases (Section 4) with our protocol. We conclude our paper and list some issues for further study in the end (Section 5).

2. Background of ID-Based Authentication and Key Establishment

For secure communication between users, authentication and key establishment is required. Successful authentication verifies user's claimed identity for the other party. Usually the authentication must happen in both directions (mutual authentication). After authentication the entities need to agree on a shared key that is used to protect the communication further on. Diffie-Hellman protocol [1] can be seen as the first key establishment protocol based on public key cryptography. However, the protocol does not provide authentication of the communicating parties, meaning that a man-in-the-middle attack is possible (an adversary between the communicating parties modifying the messages can establish separate keys with each end point). Thus, it is essential to bind authentication and key establishment together. Protocols achieving this are called authenticated key establishment (AK) [2].

Identity (ID) based cryptography (IBC) [3] builds on the basic idea that the public key of the user is based on some unique information about the user's identity, like for example an email address (string). The IBC system has become an active research topic in the recent years because of practical ID-Based encryption, signature, and key exchange applications [4, 5, 6]. In addition to using the identity as the public key the IBC

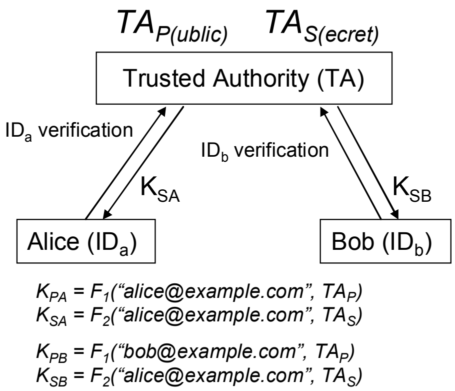
TA P(ublic)
TA S(ecret)
Trusted Authority (TA)
IDa verification
IDb verification
KSA
KSB
Alice (IDa)
Bob (IDb)
KPA = F1("alice@example.com", TAP)
KSA = F2("alice@example.com", TAS)
KPB = F1("bob@example.com", TAP)
KSB = F2("alice@example.com", TAS)

Figure 1 Trusted Authority as Private Key Generator in Identity Based Cryptography

system public parameters provided by a Trusted Authority (TA) are needed (see Figure 1).

IBC is controversial compared to the traditional certificate based systems, where a designated Certificate Authority (CA) signs (and creates) a user specific certificate, containing the user's identity and her public key. In a simple setup all the certificates are signed by a trusted CA. Every involved party has the CA's certificate for verifying the CA's signatures. When Bob wants to authenticate Alice or send encrypted information for her, he must first get her certificate and verify its validity with the CA's certificate and possibly also compare it to the revocation list. Then Bob can use Alice's public key from the certificate to encrypt information for the target. In an IBC system users do not have to get or store the public keys of the corresponding communicating parties, because they can be created based on the target's identity and the common parameters of the IBC system. However, the communicating parties must ensure that they are using the same public parameters. This is comparable to a system specific certificate instead of user specific certificate.

2.3 Implicit AKE with Key Derivation

IBC is using asymmetric key cryptography based on elliptic curves and pairings. However, identities can be bound to the symmetric keys within the key establishment protocol with key derivation functions (KDF). In simple key derivation, a root key and an identity are used as input parameters for a one-way hash function, which then produces the new key. This new key is one level lower in the key hierarchy. Binding the identity or some other parameter that is specific to the communications for which the key is used for is also called *channel binding* [7]. KDF is usually a one-way hash function, which ensures that by holding a key lower in the hierarchy, an attacker can not deduce a higher level key in the hierarchy (see Figure 3).

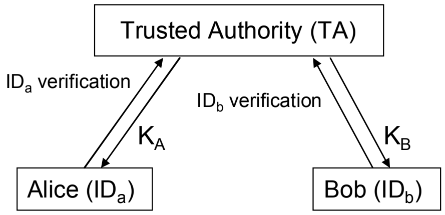

Figure 2: Trusted Authority as Shared Secret key generator in our protocol

Usually both communicating end points derive keys similarly based on some input parameters, like used algorithm, identity, nonce, etc., but it is also possible to provide keying material to different parties from different key hierarchy levels. In effect, similar usage models to the public key cryptography can be designed. Additionally, as the identity is bound to the key derivation, the mechanism provides a nice way to authenticate the identity itself and thus also the end point.

3. ID Binding with Symmetric Keys as an Implicit AKE

Let's assume that a telecom operator Opera has a TA server reachable through or integrated into the SMS (Short Message Service) gateway. Alice and Bob are both registered users of the Opera and thus have valid and unique telephone numbers. A service provider, Simon, wants to create a service into the Internet or provide secure wireless network access. This requires real user identity authentication from the clients, so that badly behaving users can be traced by appropriate legal authorities. Simon's server is a simple PC connected to the Internet with a fixed line connection and with no cellular interface. However, Simon has no resources or possibilities to start creating user accounts *by authenticating the client's identities*

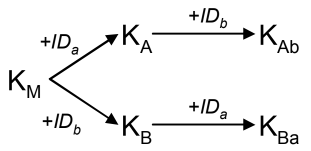

Figure 3 Key derivation hierarchy

face-to-face. Thus, Simon's only possibility is to leverage some existing user database and authentication service. Virtual operator Opera contacts Simon and offers a simple win-win deal as follows. Opera provides a symmetric secret key K_S for Simon and asks Simon to install it into his PC, but also to keep it secret. Then Opera explains Simon that he can start his service and authenticate users with their phone number (or some random pseudonym in the users request message) as their user name and a PIN code as their password. Opera explains that the password PIN code is based on the K_S and user's phone number (or pseudonym) and that he does not need to create the user database in advance. Operator Opera also explains that if the key K_S is compromised, Simon can always ask a new key from the operator, and that in fact the lifetime of the key K_S is one month (as an example), after which a new key must be installed for the service. Simon is happy as he can start providing the service right away without a need to establish the user account base from the scratch.

To create the environment as outlined above, we use the idea of Identity (ID) binding with symmetric keys and create a novel AKE protocol with possibilities to use either sender's symmetric key and receiver's identity or receiver's symmetric key and sender's identity as basis for the key establishment. Although, the scheme is very simple from a cryptographic function point of view, we want to show that key derivations can be used to achieve similar constructs as with IBC and Kerberos [8] with less complexity.

The model requires a common trusted authority (TA, see Figure 2, operator Opera in our example use case), and that each node in the system must have a unique identity (phone number in our example). Each communicating party must be able to mutually authenticate with the TA and agree a long enough symmetric key (SMS messages). Further on, the aim is that two nodes with a common TA can mutually authenticate and send integrity and/or confidentiality protected packets to each other (Simon and his customers). To achieve this, we utilize key derivation with identity binding based either on sender's key or receiver's key (see Figure 3).

3.1 Sender's ID based AKE

In the Sender's ID (SID) based AKE scheme (see Figure 4), Alice asks the TA to derive herself a session key between herself and Bob. Alice sends a *Receiver Key Request* message through a secure channel for TA along with Bob's ID. TA authenticates the request and finds out if Bob has registered. If Bob has registered to the system, TA derives receiver key K_{Ba} for Alice and sends it to her via the secure authenticated channel.

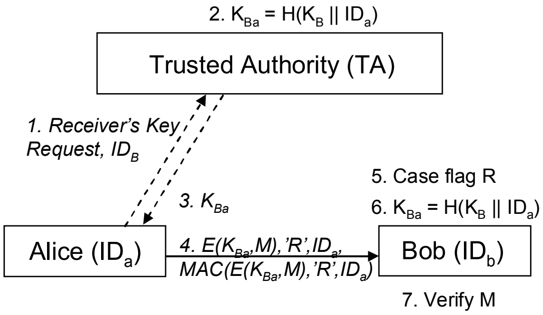
2. $K_{Ba} = H(K_B \,||\, ID_a)$
Trusted Authority (TA)
1. Receiver's Key Request, ID_B
3. K_{Ba}
5. Case flag R
6. $K_{Ba} = H(K_B \,||\, ID_a)$
Alice (ID_a)
4. $E(K_{Ba},M)$,'R',ID_a, $MAC(E(K_{Ba},M)$,'R',$ID_a)$
Bob (ID_b)
7. Verify M

Figure 4: Sender's ID-Based AKE

$$K_{Ba} = H(K_B \,||\, ID_a) \qquad (2)$$

Alice then sends integrity protected and encrypted message for Bob over an insecure channel along with her own ID and a flag (S) in the message noting that the SID based AKE is used. When Bob gets the message he takes Alice's ID and his own shared key with TA K_B and derives K_{Ba} as in *(2)*.

To continue with our example use case, Alice finds out about Simon's service on the Internet and decides to try it out. She browses to the URL of Simon's service and finds out that it requires user authentication. Alice then sends an SMS to the Opera's TA along with Simon's service ID (ID_S, sender ID based AKE). The TA then checks the incoming phone number, finds out Simon's service key K_S and derives PIN code (i.e. password and optionally a pseudonym) and sends it back to Alice. Alice types her phone number as the user name and the received PIN code as the password on the Simon's service login web page over a secure connection (e.g. TLS with server certificate).

Simon's server receives a login request with Alice's phone number. Using the service key K_S, the server then derives a new key K_{SA} based on (2) and compares the result with the PIN code (using as many digits from the K_{SA} as necessary) and finds out that they match. The server also checks from the user database if the user with this phone number has logged into the service before in case some user specific customization parameters would have been configured. Not at this time. Thus, Alice's login is an authenticated registration to the service. The server then stores the phone number into the user database along with any service customization parameters Alice has selected.

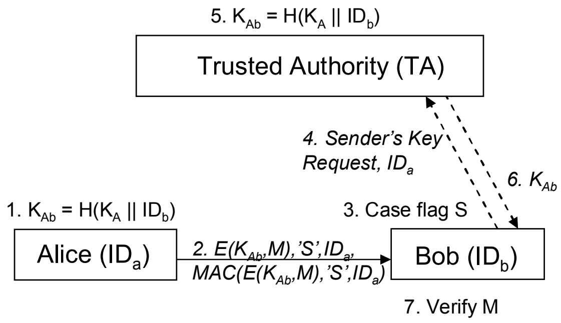
5. $K_{Ab} = H(K_A \mid\mid ID_b)$
Trusted Authority (TA)
4. Sender's Key Request, ID_a
6. K_{Ab}
1. $K_{Ab} = H(K_A \mid\mid ID_b)$
3. Case flag S
Alice (ID_a)
2. $E(K_{Ab},M)$,'S',ID_a, $MAC(E(K_{Ab},M)$,'S',$ID_a)$
Bob (ID_b)
7. Verify M

Figure 5: Receiver's ID-Based AKE

After a couple of days Alice tries to log in again, but notices that she has forgotten the PIN code for the service. She then sends a new SMS message with the service provider's ID to the TA. After a month has passed, Alice logs in again as usual with the same phone number and PIN code she has in her SMS messages inbox. However, the service informs Alice to get a new PIN code with the SMS message because the previous PIN code has become too old.

For another Internet service, Simon's wants to use the same authentication method, but wants to restrict the users to a certain country only in the beginning. Thus, he makes a deal with the operator Opera that only users living in the specific country area are allowed to get PIN codes for the service. Operator Opera then filters out PIN code requests from users that are not registered into the specific country.

Alice notices that there are a huge number of services utilizing the SMS based authentication service. Thus, Alice is able to get a PIN code for the services with an SMS message indicating all the service IDs of the services Alice wants to use.

3.2 Receiver's ID based AKE

In a Receiver's ID (RID) based AKE scheme (see Figure 5), Alice takes her own symmetric key K_A and derives a new key K_{Ab} for Bob based on Bob's identity. Using a one-way hash function *H*, K_A, and Bob's identity ID_b as input parameters Alice gets a proper key for Bob, K_{Ab} (|| *denotes concatenation and H produces the same number of bits as the key length for simplicity*).

$$K_{Ab} = H(K_A \mid\mid ID_b) \qquad (1)$$

Using the resulting key Alice sends integrity protected and encrypted message over an insecure channel for Bob along with her own identity and a flag

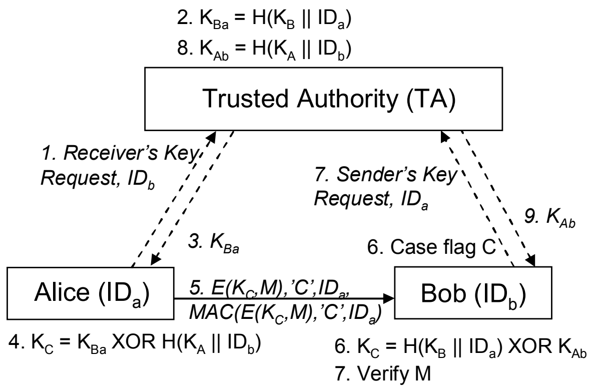
2. $K_{Ba} = H(K_B \mid\mid ID_a)$
8. $K_{Ab} = H(K_A \mid\mid ID_b)$
Trusted Authority (TA)
1. Receiver's Key Request, ID_b
7. Sender's Key Request, ID_a
3. K_{Ba}
9. K_{Ab}
6. Case flag C
Alice (ID_a)
5. $E(K_C,M)$,'C',ID_a, $MAC(E(K_C,M)$,'C',$ID_a)$
Bob (ID_b)
4. $K_C = K_{Ba}$ XOR $H(K_A \mid\mid ID_b)$
6. $K_C = H(K_B \mid\mid ID_a)$ XOR K_{Ab}
7. Verify M

Figure 6 Combined Sender and Receiver ID Based AKE

(R) in the message that indicates the usage of RID based AKE. Once Bob gets a message from Alice, he sends *Receiver's ID Key Request* along with Alice's ID for the TA through a secure authenticated channel between Bob and the TA. After TA has authenticated the request, it checks if Alice has registered and finds out that she is. Since TA knows the shared secret with Alice it can derive the same key K_{Ab} for Bob (see Figure 2). TA sends the key through the secure channel for Bob. Bob authenticates the received message from TA and gets the key, which it uses to authenticate the message from Alice, provided that Alice's shared key with the TA has not been compromised.

To continue with our example use case, Simon now wants to extend his service offering with a new push style services to reach his current and new customers more efficiently. Users, like Alice and Bob, willing to receive secure and personalized offers from the service providers, like Simon, register to the SMS gateway and while they get their shard secrets they also register their preferences to the push services directory. Then Opera (the SMS provider) provides a list of identities/phone numbers that the service provider is allowed to use along with the derived shared secrets. Simon creates a user tailored special offer push message secured with corresponding target user's shared secret. Alice gets the push message from Simon and is able to authenticate the message by using her own shared secret and the Simon's service identity and thus verify that the offer is valid (e.g. user can verify the source of the offer as well be sure that it was targeted to her only). User can use the session key further on when authenticating to the Simon's service portal and buying the product that the personalized push advertisement was offering for her.

3.3 Combined Sender and Receiver ID-Based AKE

To increase the security of the key we can use both sender and receiver IDs in the key derivation function (see Figure 6).

$$K_C = K_{Ba}\ xor\ K_{Ab} \qquad (3)$$

This efficiently requires both the sender and the receiver contact the TA for the message delivery and authentication. In other words, Alice and Bob need to contact the TA to derive the combined key because they are not able to derive the other key based on their own keys.

3.4 One time PIN

In case the service provider Simon wants to utilize one-time-passwords, and thus force Alice to get a new PIN code (or password) for every new session, the TA can issue K_{Ba} for the user based on the following formula.

$$K_{Bai} = H(H(K_B \mid\mid i) \mid\mid ID_a) \qquad (4)$$

where i is long enough serial number starting from pre-defined value k. Both the service provider Simon and the TA agree on the value k, at the same time they agree on the shared key. However, service provider needs then to keep counter values for each authenticated user in their profiles, which makes this scheme less interesting.

3.5 Distributed TA with common master key

Changing the key derivation function in the TA, it is possible to derive user specific keys based on a master key K_M. Figure 7 shows how the two TAs share the key but use it to derive user specific keys based on the user's identity. The problem with the distributed TA is that the master key needs to be the same for each of the distributed TAs.

To continue with our example use case, the operator Opera could provide the same master key for multiple SMS centers, or even that the multiple operators in one area, for example a city or country all agree on a common master key. This way the service provider Simon would not know customer Alice's or Bob's operator. Alice could be from another operator than Bob.

4. Additional Use Cases

In this section we list some additional use cases that utilize our identity based implicit authentication and key establishment with symmetric keys protocol.

4.1 Pre-Shared Key TLS with Sender and Receiver ID-Based AKE

Pre-shared key TLS [9] (PSK-TLS) describes shared key cipher suites for TLS protocol [10]. TLS is an AKE protocol used in many applications and services, like for example in secure HTTP. However, PSK TLS does not support either Sender or Receiver ID based key derivation schemes. Adding support for this in PSK-TLS would allow setups in which for example a centralized Operations & Management (O&M) server would contain a master secret and all clients for the O&M system would be using Sender ID based AKE, with pre-configured K_{Ba} (e.g. steps 1, 2, and 3 skipped in Figure 4). This would allow the administrators to add clients to the O&M system, without the need to add/change configurations in the O&M server for each client, but still having separate keys for each client instead of shared client keys.

4.2 IP Packet Authentication

In the future Internet, DNS may be considered to be secure enough [11] to act as a TA in the Internet. Consider an *example.com* domain with firewall that has a shared key K_B with the corresponding DNS domain master server. The firewall wants to authenticate IP packets for multiple hosts inside example.com domain (behind the firewall). Thus, we can apply Sender's ID based AKE with DNS as the TA for all incoming IP packets to the firewall. The firewall could also be an access router for a wireless access network.

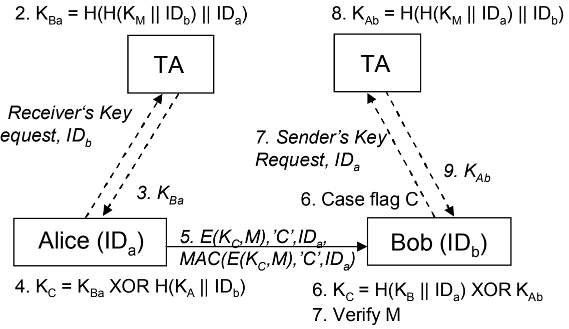
2. $K_{Ba} = H(H(K_M \mid\mid ID_b) \mid\mid ID_a)$
8. $K_{Ab} = H(H(K_M \mid\mid ID_a) \mid\mid ID_b)$
TA
TA
1. Receiver's Key Request, ID_b
7. Sender's Key Request, ID_a
9. K_{Ab}
3. K_{Ba}
6. Case flag C
Alice (ID_a)
5. $E(K_C,M)$,'C',ID_a, $MAC(E(K_C,M),'C',ID_a)$
Bob (ID_b)
4. $K_C = K_{Ba}$ XOR $H(K_A \mid\mid ID_b)$
6. $K_C = H(K_B \mid\mid ID_a)$ XOR K_{Ab}
7. Verify M

Figure 7 TA distribution with one master key

If the master DNS server configures the Time To Live (TTL) value for all hosts in example.com domain as 0, it means that all DNS queries end up for the example.com master DNS server (e.g. no caching), which can then provide K_{Ba} for each client A. Here, the master DNS server must know what the Sender's ID is. Thus, we need an extension to carry the sender ID in the DNS query itself. One way to achieve this would be to include the Sender ID as an additional prefix for the DNS name being queried. For example: *IPaddr-X-Y-Z-V.www.example.com*, where the sender's IP address is *X.Y.Z.V.* Then the master DNS server for domain example.com may or may not support this extension. If it supports it returns the IP address of www.example.com along with the K_{Ba} key for the sender, which uses it to create an authentication header for the packet towards the domain.

When the firewall gets an incoming packet with, it takes the source address, its own key and derives the shared secret K_{Ba} with (2) and uses it to verify the authentication header. The symmetric key derivation procedure itself is fast (one-way hash function) and thus storing the key into memory is not necessary.

We can also extend the KDF to include the receiver's IP address. In this case the firewall computes the key with the following function:

$$K_{Ba} = H(H(K_B \,||\, ID_d) \,||\, ID_a) \qquad (5)$$

where ID_d is the destination IP address of the corresponding server in the example.com domain.

5. Related Work

Dutta et al. [2] provide a nice overview of different key establishment protocols. In their paper they divide key establishment protocols into two categories, namely certificate based and ID based. Further on they divide the protocols in two-party, three-party, group, and tree based group key establishment protocols. Two-party key establishment protocols include ID based key establishment protocols based on pairings. Chen et al. [12] provide a very comprehensive comparison and overview of ID-Based key establishment protocols based on pairings. They also evaluate the efficiency of the different protocols. Pairing based IBC protocols are utilizing supersingular elliptic curve cryptography with the assumption that Bilinear Diffie-Hellman (BDH) problem is considered hard (e.g. given P, aP, bP, cP computing $\hat{e}(P,P)^{abc}$ is hard). Dutta et al. have also a survey paper on pairing based cryptographic protocols [13]. For more information about IBC systems, readers should refer to "A survey on ID-Based Cryptographic Primitives" from Gorantla et al. [14]

Self-certified keys and signature scheme is an alternative for traditional certificate based systems, because the sender's public key is extracted from the trusted third party's (e.g. a CA) signature for the senders identity. B. Brumley [4] presents an application of self-certified and identity based certificates with efficient three-term simultaneous elliptic scalar multiplication, where the signature scheme is based on Nyberg-Rueppel signatures by a trusted third party [5].

Shih-I-Huang [15] presents a simple key derivation based on node identities to reduce the number of keys needed for a PIKE keying scheme for sensor networks [16]. The basic idea there is that a one-way hash function dependency exists between two keys of two sensors. The other sensor knows how to create a key for the other based on the target node's number in the PIKE scheme.

Kerberos [8] is not using key derivation, but is in effect closely related to channel binding mechanisms with symmetric keys. Kerberos uses tickets, which include an encrypted session key for the authenticator. The user gets the ticket along with a session key. She provides the ticket to a server, which decrypts the ticket and gets the same session key as what the user has. This way Kerberos is a key distribution protocol without explicit key derivation. However, Kerberos could be extended in such a way that the session key itself is based on some KDF function that binds the keys to the right context (like users' identity).

Related to our IP packet authentication use case with secure DNS, Candolin, Lundberg, and Kari [17] present a packet level authentication scheme for military networks based on public keys in [17].

5.1 Trusted Authority as IBC Public Key Generator

In asymmetric key IBC with a Private Key Generator (PKG) or sometimes called Key Generation Center (KGC) is trusted by all users (e.g. a Trusted Authority, TA) and is responsible for the generation of the user's corresponding private keys. Each user then gets its private key from the PKG, but also the common parameters used to create the public keys based on the receiver's identity.

With the early ID-Based authentication and key establishment protocols key escrow is possible by the PKG, meaning that the PKG can deduce the key used to secure the communication by simply wiretapping the conversation (PKG knows how to create the corresponding secret keys based on the used identities). However, Chen and Kudla [18] have developed a protocol in which the key escrow feature can be turned off. They also provide an extension to their protocol, which allows users under different PKGs to agree a key together. Later more efficient schemes have been proposed [19], abut also some security considerations for all these key escrow disabling schemes [20].

Gentry and Silverberg introduced a Hierarchical ID-Based Encryption (HIBE) scheme [21], which is a generalization of ID-Based encryption that reflects organizational hierarchies. This lessens the burden from a single PKG to multiple PKGs. An identity at level k of the hierarchy tree can issue private keys to its descendant identities, but cannot decrypt messages intended for other identities. Boneh et al. [22] described an improved HIBE scheme, which consumes fewer bits than the Gentry and Silverberg one. Boneh et al. also describe a mechanism on how to provide *forward security* for the ID-Based cryptosystem.

Balfanz et al. describe secret handshakes from Pairing-Based Key establishments [23]. Their aim is to provide an analogical secret society (for example CIA) identification handshake with the AKE protocol. They describe how IBC with pairing can be used to establish secure sessions between two entities based on the IBC Trusted Authority (TA) parameters and the peer's pseudonym or even based on the peer's claimed role. Instead of publicly meaningful identities, they use pseudonyms and pre-defined roles for the users. By using pseudonyms instead of public identities they loose the best feature within IBC, namely the binding of the real identity with the public key. In case the handshake fails because the peers used different roles for each other, some information may be leaked (e.g. the peer is not using this role for this particular key establishment).

Burnett et al. describe in their paper how biometric identity information can be used as the identity information with ID-Based signature scheme [24]. They address the problems of fuzziness with biometric identity measurement as well.

HP Laboratories have done research in the area of IBC based applications, for example within the area of role based secure message service, privacy, and identity management for the health care systems etc. [25, 26].

6. Conclusion

We created a simple symmetric key based AKE protocol that binds sender and/or receiver identities to the key establishment and thus provides implicit authentication of the identities based on the trusted third party. We provided new and interesting use cases, especially one for telecom operators that can utilize our protocol and the SMS as a good enough confidential channel for communications where the operator is a trusted authority. Another use case was IP packet authentication based on trusted DNS.

Our sender and receiver ID based AKE protocol with symmetric keys has overlapping applications with public key ID-Based AKE. Both of them are useful, with slightly different setups as is also the case when comparing symmetric key and public key cryptography together. We believe that the term ID-Based Cryptography AKE used only with public key cryptography may be a slightly confusing term as identity based AKE can also be done with symmetric keys. The drawback with our protocol is that it requires more interactions with the TA than the asymmetric key IBC AKE scheme. The quantification of this is left for further study as well as the detailed security analysis and security proof of our protocol.

Acknowledgment

We would like to thank prof. Nadarajah Asokan, prof. Kaisa Nyberg, Maarit Hietalahti, and Vesa Vaskelainen for valuable comments on this paper.

References

[1] W. Diffie, M. Hellman, "New Directions in Cryptography", In IEEE Transactions on Information Theory, IT-22 (6), pp. 644-654, 1976

[2] Ratna Dutta, Rana Barua, "Overview of Key establishment Protocols", 2005, Cryptology ePrint Archive: Report 2005/289, URL: http://eprint.iacr.org/2005/289.ps (referenced 2007-04-13)

[3] Ad Shamir, "Identity-based Cryptosystems and Signature Schemes", In proceedings of Crypto 1984, LNCS 196, pp. 47-53, Springer-Verlag, 1984

[4] B. Brumley, "Efficient Three-Term Simultaneous Elliptic Scalar Multiplication with Applications ", Proceedings of Norsec 2006

[5] Giuseppe Ateniese and Breno de Medeiros, "A provably secure Nyberg-Rueppel signature variant with applications", Technical Report 93, Cryptology ePrintArchive, 2004.

[6] D. Boneh, M. Franklin, "Identity-based encryption from the Weil pairing", SIAM Journal of Computing, 32(3):568-615, 2003

[7] B. Aboba, D. Simon, J. Arkko, P. Eronen, and H. Levkowetz, "Extensible Authentication Protocol (EAP) Key Management Framework", draft-ietf-eap-keying-14.txt, Internet draft (work in progress), 2006-06-27.

[8] J. Steiner, C. Neuman, and J.I Schiller, "Kerberos: An Authentication Service for Open Network Systems, " in Proc. Winter USENIX Conference, Dallas (1988).

[9] P. Eronen, H. Tschofenig, "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", IETF RFC4279, December 2005.

[10] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", IETF RFC2246, January 1999.

[11] IETF DNS Extensions (dnsext) Working Group, URL: http://www.ietf.org/html.charters/dnsext-charter.html (referenced 2007-04-13)

[12] L.Chen, Z.Cheng, N.P. Smart, "Identity-based Key Agreement Protocols from Pairings", Cryptology ePrint Archive: Report 2006/199, URL: http://eprint.iacr.org/2006/199 (referenced 2007-04-13)

[13] Ratna Dutta and Rana Barua and Palash Sarkar, "Pairing-Based Cryptographic Protocols: A Survey", Cryptology ePrint Archive: Report 2004/064, URL: http://eprint.iacr.org/2004/064 (referenced 2007-04-13)

[14] M. Choudary Gorantla and Raju Gangishetti and Ashutosh Saxena, "A Survey on ID-Based Cryptographic Primitives", Cryptology ePrint Archive: Report 2005/094, URL: http://eprint.iacr.org/2005/094 (referenced 2007-04-13)

[15] S.I. Huang. "Adaptive random key distribution schemes for wireless sensor networks", In Proceedings of the International Workshop on Advanced Developments in Software and Systems Security, 2003.

[16] A Haowen, Chan Perrig. "Pike: Peer intermediaries for key establishment in sensor networks", In INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE , vol.1, no.pp. 524- 535 vol. 1, 13-17 March, 2005.

[17] K. Candolin, J Lundberg, and H Kari. "Packet level authentication in military networks". In Proceedings of the 6th Australian Information Warfare & IT Security Conference, Geelong, Australia, November 2005.

[18] L. Chen, C. Kudla, "Identity Based Authentication Key Agreement Protocols from Pairings", Cryptology ePrint Archive: Report 2002/184, URL: http://eprint.iacr.org/2002/184 (referenced 2006-10-15)

[19] N. McCullagh, P.S.L.M. Barreto, "A New Two-Party identity-Based Authenticated Key Agreement", In proceedings of CT-RSA 2005, LNCS 3376, pp. 262-274, Springer-Verlag, 2005.

[20] Kim-Kwang Raymond Choo, "Revisit Of McCullagh--Barreto Two-Party ID-Based Authenticated Key Agreement Protocols", Cryptology ePrint Archive: Report 2004/343, URL: http://eprint.iacr.org/2004/343 (referenced 2007-04-13)

[21] Craig Gentry and Alice Silverberg, "Hierarchical ID-Based Cryptography", Cryptology ePrint Archive: Report 2002/056, URL: http://eprint.iacr.org/2002/056 (referenced 2006-10-15)

[22] D. Boneh, E.-J. Goh, and X. Boyen, "Hierarchical Identity Based Encryption with Constant Size Ciphertext", Cryptology ePrint Archive: Report 2002/015, URL: http://eprint.iacr.org/2005/015.pdf (referenced 2007-04-13)

[23] Balfanz, D., Durfee, G., Shankar, N., Smetters, D., Staddon, J., and Wong, H. 2003. "Secret Handshakes from Pairing-Based Key Agreements". In Proceedings of the 2003 IEEE Symposium on Security and Privacy (May 11 - 14, 2003). SP. IEEE Computer Society, Washington, DC, 180.

[24] A. Burnett, A. Duffy, T. Dowling, "A Biometric Identity Based Signature Scheme", Cryptology ePrint Archive: Report 2004/176, URL: http://eprint.iacr.org/2004/176 (referenced 2007-04-13)

[25] M. Casassa Mont, P. Bramhall, "IBE Applied to Privacy and Identity Management," Hewlett-Packard Laboratories, technical report HPL-2003-101, 2003

[26] M. Casassa Mont, P. Bramhall, C. R. Dalton, K. Harrison, "A Flexible Role-based Secure Messaging Service: Exploiting IBE Technology in a Health Care Trial," Hewlett-Packard Laboratories, technical report HPL-2003-21, 2003.