Comparison of Existing Key Establishment Protocols

Boyeon Song and Kwangjo Kim

Information and Communications University (ICU)
58-4 Hwaam-dong, Yusong-gu, Taehyon, 305-732, S. Korea
{bysong, kkj}@icu.ac.kr

Abstract

This paper classifies existing key establishment protocols according to cryptographic techniques, models of environment used, and the number of pass. The protocols’ messages are described briefly and its properties are compared in four groups; key transport protocols using symmetric techniques, key transport protocols using asymmetric techniques, key agreement protocols using symmetric techniques, and key agreement protocols using asymmetric techniques.

1 Introduction

Cryptographic keys need to be established to the communicating parties prior to secure communications for security services which is based on cryptographic mechanisms like data encipherment, message authentication, and digital signatures. Key establishment protocol is a process whereby a shared secret key becomes available to participating entities, for subsequent cryptographic use. It is broadly subdivided into key transport and key agreement protocol[11]. In key transport protocols, a key is created by one entity and securely transmitted to the other entity. In key agreement protocols, both entities contribute information to generate the shared secret key[4].

Key establishment protocol employs symmetric or asymmetric key cryptography. A symmetric cryptographic technique is a system involving two transformations - one for the initiator and one for the responder - both of which make use of the same secret key[11]. In this system the two entities previously possess common secret information, so the key management problem is a crucial issue. An asymmetric cryptographic technique is a system involving two related transformations - one defined by a public key (the public transformation), and another defined by a private key (the private transformation) - with the property that it is computationally infeasible to determine the private transformation from the public transformation[11]. In this system the two entities share only public information that has been authenticated[5].

Key establishment protocol can be grouped into three models by the nature of the communication links, the trust relationships involved, and the cryptographic techniques used: point-to -point key establishment, key establishment within one domain, and key establishment between domains.

This paper classifies key establishment protocols, describes existing protocols, and compare them according to some requirements needed for secure key establishment. The remaining of this paper is organized as follows. Section 2 explains general models for key establishment environment, basic requirements of key establishment, and notations common to all the protocols. Section 3 classifies existing key transport protocols as cryptographic techniques, models of environment used, and the number of pass, and describes the briefs of the protocols’ messages. Section 4 classifies and describes key agreement protocols in the same way. Section 5 compares key establishment protocols according to the requirements described in section 2. Finally, Section 6 makes concluding remarks.
2 Preliminary

2.1 Models for Key Establishment Environment

Key establishment protocols are influenced by the nature of the communication links, the trust relationships involved and the cryptographic techniques used; the entities may either communicate directly or indirectly, may belong the same or different security domains, and may or may not use the services of a trusted authority. The environments for the establishment of keys are illustrated[14].

2.1.1 Point-to-Point Key Establishment

Point-to-point key establishment is the basic mechanism of every key establishment scheme. It is required that the entities have to share a key so that further keys may be established directly between the entities[7, 14].

- When the entities use symmetric techniques,
  the two parties involved already share a key that can be used to protect the keying material to be established.

- When the entities use asymmetric techniques,
  each of the parties has a public key with its associated private key, and that the authenticated public key is known to the other party; for data integrity of data origin authentication, the recipient requires the sender’s corresponding public key certificate, and for confidentiality, the sender does a public key certificate of the intended recipient.

2.1.2 Key Establishment Within One Domain

Key establishment within one domain employs a single Trusted Third Party (TTP) for the entire system. This authority may offer key management services such as the generation, certification, distribution or translation or keying material[7, 14].

- When the entities use symmetric techniques,
  a sender and a receiver are required to share keys with the authority. Key establishment then is initiated in one of two ways.

  - **Key Distribution Center** (KDC) : By one entity asking a KDC to generate a key for subsequent distribution. The KDC then either distributes the key directly to both entities, or it sends it back to the initiator, who forwards it to the other entity.

  - **Key Translation Center** (KTC) : By one entity generating the key and sending it to a KTC. The KTC may either forward directly it to the other entity, or send it back to the first entity, who forwards it to the second entity.

- When the entities use asymmetric techniques,
  each entity may need to contact its authority to get an appropriate public key certificate. Then the TTP typically is called Certification Authority (CA).

  Key establishment for large systems usually is organized in a hierarchical way. Security Domain is a group of entities served by one authority[7, 14].
2.1.3 Key Establishment Between Domains

This model involves entities belonging to different security domains which share at least one cryptographic technique. Each security domain has its own security authority[7],

- When the entities use symmetric techniques and do not have access to a common directory service that offers public key certificates, each entity shall contact its respective authority to get its partner’s public key certificate. The authorities of A and B may exchange the public key certificates of entities A and B and forward them to A and B, respectively.

- When the entities use asymmetric techniques, At least one of them has to contact its authority to receive a secret key for communication. The authorities then establish a common secret key to be used by the entities. This key may be distributed by one authority to both entities using the other authority as a KTC, or via one of the entities which is responsible for forwarding the key to the other entity.

In situations where the authorities of A and B neither have a mutual trust relationship nor a direct communications path, they have to involve one of more additional TTPs until a chain of trust is established.

2.2 Requirements

Required properties for a key establishment protocol may include the followings[7, 16],

- **Entity Authentication of A to B**: the assurance of the identity of A for B

- **Implicit Key Authentication from A to B**: the assurance for B that A is the only other entity that can possibly be in possession of the correct key.

- **Key Confirmation from A to B**: the assurance for B that A is in possession of the correct key.

- **Explicit Key Authentication from A to B**: the assurance for B that A is the only other entity that is in possession of the correct key. (Implicit key authentication from A to B and key confirmation from A to B together imply explicit key authentication from A to B)

- **Efficiency**: Considerations include the number of message exchanges (passes) required, the number of bits transmitted, the complexity of computations required by each party, and possibility of precomputations.

- **Key Freshness**: a guarantee that the established keying material is new, as opposed to the reuse of old keying material.

- **Key Control**: the ability to choose the key or the parameters used in the key computation.

Security requirements are concerned with the confidentiality of a key, modification and replay detection and the detection of substitution. In practical implementations of key establishment protocols the key data may be need to further processing prior to being used for cryptographic communication. Key establishment protocols may contain optional data, depending upon the specific application. One such possible application is message authentication[15].
2.3 Notation

The following notations are used throughout this paper.

<table>
<thead>
<tr>
<th>notation</th>
<th>meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_X$</td>
<td>distinguishing identifier of entity $X$</td>
</tr>
<tr>
<td>KDC</td>
<td>Key Distribution Center</td>
</tr>
<tr>
<td>KTC</td>
<td>Key Translation Center</td>
</tr>
<tr>
<td>CA</td>
<td>Certification Authority</td>
</tr>
<tr>
<td>$T$</td>
<td>distinguishing identifier of KDC or KTC</td>
</tr>
<tr>
<td>$K$</td>
<td>secret key for a symmetric cryptosystem</td>
</tr>
<tr>
<td>$K_X$</td>
<td>secret symmetric session key chosen by entity $X$</td>
</tr>
<tr>
<td>$K_{XY}$</td>
<td>secret key associated with the entities $X$ and $Y$</td>
</tr>
<tr>
<td>$r$, $m$</td>
<td>random number</td>
</tr>
<tr>
<td>$r_X$</td>
<td>random number issued by entity $X$</td>
</tr>
<tr>
<td>$t/n$</td>
<td>time stamp or a sequence number</td>
</tr>
<tr>
<td>$t_X/n_X$</td>
<td>time stamp or a sequence number issued by entity $X$</td>
</tr>
<tr>
<td>$L$</td>
<td>validity period (life time)</td>
</tr>
<tr>
<td>$TVP$</td>
<td>time variant parameter</td>
</tr>
<tr>
<td>$TVP_X$</td>
<td>time variant parameter issued by entity $X$</td>
</tr>
<tr>
<td>$E_K(Z)$</td>
<td>encipherment of data $Z$ with a symmetric algorithm using the key $K$</td>
</tr>
<tr>
<td>$D_K(Z)$</td>
<td>decipherment of data $Z$ with a symmetric algorithm using the key $K$</td>
</tr>
<tr>
<td>$P_X(Z)$</td>
<td>entity $X$'s public encipherment transformation of data $Z$</td>
</tr>
<tr>
<td>$S_X(Z)$</td>
<td>entity $X$'s private signature transformation on data $Z$</td>
</tr>
<tr>
<td>$MAC_K(Z)$</td>
<td>message authentication code (MAC) on data $Z$ using the key $K$</td>
</tr>
<tr>
<td>$f$</td>
<td>key generating function</td>
</tr>
<tr>
<td>$X, Y$</td>
<td>concatenation of data items $X$ and $Y$ in that order</td>
</tr>
<tr>
<td>$Cert_X$</td>
<td>entity $X$’s public key certificate</td>
</tr>
<tr>
<td>$a$, $b$</td>
<td>entity $A$ and $B$’s static private keys; $a, b \in_R [1, q - 1]$</td>
</tr>
<tr>
<td>$z_A$, $z_B$</td>
<td>entity $A$ and $B$’s static public keys; $z_A = g^a \mod p$, $z_A = g^b \mod p$</td>
</tr>
<tr>
<td>$x$, $y$</td>
<td>entity $A$ and $B$’s ephemeral private keys; $x, y \in_R [1, q - 1]$</td>
</tr>
<tr>
<td>$p$</td>
<td>a large prime</td>
</tr>
<tr>
<td>$q$</td>
<td>a prime divisor of $p - 1$</td>
</tr>
<tr>
<td>$g$</td>
<td>an element of order $q$ in $Z_p^*$ which is shared public by all the entities</td>
</tr>
<tr>
<td>$H$</td>
<td>cryptographic hash function</td>
</tr>
<tr>
<td>$A \rightarrow B : Z$</td>
<td>$A$ sends $Z$ to $B$</td>
</tr>
<tr>
<td>$A \Longrightarrow B : Z$</td>
<td>$A$ sends $Z$ to $B$ over a secure channel</td>
</tr>
<tr>
<td>$A : Z$</td>
<td>$A$ executes $Z$</td>
</tr>
</tbody>
</table>

For the implementation of the protocols specified in this paper[15, 16], it is assumed that

- When the entities use symmetric techniques
  - Point-to-Point
    - A key $K_{AB}$ is shared by $A$ and $B$.
  - TTP (KDC, KTC)
    - There is a TTP $T$, with which $A$ and $B$ share secret keys, $K_{AT}$ and $K_{BT}$ respectively. $T$ shall be able to generate or otherwise acquire a key $K$.

- When the entities use asymmetric techniques

  1. The entities are aware of each other’s claimed identities. This may be achieved by the inclusion of identifiers information exchanged between the two entities, or it may be apparent from the context of the use of the mechanism. (Verifying the identity means to check that a received identifier field agrees with some known value or prior expectation.)
2. If a public key is registered with an entity then that entity shall make sure that the entity who registers the key is possession of the corresponding private key.

3. The authenticated public keys of entities $A$ and $B$, $z_A$ and $z_B$, are known between $A$ and $B$.

3 Key Transport

Key transport protocols make use of symmetric or asymmetric encipherment techniques.

3.1 Symmetric techniques

This section presents the brief protocols of key transport protocols based on symmetric techniques. They are subdivided into protocols using three environments models: Point to Point, Key Distribution Centre (KDC) and Key Translation Centre (KTC). A point-to-point environment exists when two entities already share a key that can establish further keys. If two entities wish to communicate with each other using only symmetric techniques but do not currently share such a key, they shall make use of a KDC or KTC.

3.1.1 Point-to-Point

1. One-Pass

- **KTS 1:** Mechanism 1 of [15]
  
  $A \rightarrow B : TVP$
  
  $K = f(K_{AB}, TVP)$

- **KTS 2:** Mechanism 2 of [15]
  
  $A \rightarrow B : E_{K_{AB}}(K)$

- **KTS 3:** Mechanism 3 of [15]
  
  $A \rightarrow B : E_{K_{AB}}(t/n, I_B, K)$

2. Two-Pass

- **KTS 4:** Mechanism 4 of [15]
  
  $A \leftarrow B : r_B$
  
  $A \rightarrow B : I_B, I_A, r_B, MAC_{K_{AB}}(I_B, I_A, r_B, K)$

3. Three-Pass

- **KTS 5:** Authenticated Key Exchange Protocol 2 (AKEP2) [11]
  
  $A \rightarrow B : r_A$
  
  $A \rightarrow B : I_B, I_A, r_B, r_A, MAC_{K_{AB}}(I_B, I_A, r_B, r_A)$
  
  $A \rightarrow B : I_A, r_B, MAC_{K_{AB}}(I_A, r_B)$
  
  $K = f(r_B, K_{AB})$

- **KTS 6:** Authenticated Key Exchange Protocol 1 (AKEP1) [11]
  
  $A \rightarrow B : r_A$
  
  $A \rightarrow B : I_B, I_A, r_B, r_A, r'_B, K \oplus f(r'_B, K'_{AB}), MAC_{K_{AB}}(I_B, I_A, r_B, r_A, r'_B, K \oplus f(r'_B, K'_{AB}))$
  
  $A \rightarrow B : I_A, r_B, MAC_{K_{AB}}(I_A, r_B)$
  
  $K = f(r_B, K'_{AB})$

- **KTS 7:** Shamir’s no-key protocol
  
  $A \rightarrow B : K^{r_A} \mod p$
  
  $A \rightarrow B : (K^{r_A})^{r_B} \mod p$
  
  $A \rightarrow B : (K^{r_A r_B})^{r_A^{-1}} \mod p$
  
  $B : K = (K^{r_B})^{r_A^{-1}} \mod p$
3.1.2 KDC

1. Three-Pass

- **KTS 8**: Mechanism 7 of [15]
  
  \[
  A \rightarrow T : I_B \\
  A \leftarrow T : E_{K_{AT}}(K, I_B), E_{K_{BT}}(K, I_A) \\
  A \rightarrow B : E_{K_{BT}}(K, I_A)
  \]

- **KTS 9**: Mechanism 10 of [15]
  
  \[
  A \rightarrow T : E_{K_{AT}}(t_A/n_A, I_B) \\
  A \leftarrow T : E_{K_{AT}}(t_T/n_T, K, I_B) \\
  T \rightarrow B : E_{K_{BT}}(t'_T/t'_T, K, I_A)
  \]

2. Four-Pass

- **KTS 10**: Mechanism 8 of [15]
  
  \[
  A \rightarrow T : TP_A, I_B \\
  A \leftarrow T : E_{K_{AT}}(TP_A, K, I_B), E_{K_{BT}}(t_T/n_T, K, I_A) \\
  A \rightarrow B : E_{K_{BT}}(t_T/n_T, K, I_A), E_{K}(t_A/n_A, I_B) \\
  A \leftarrow T : E_{K}(t_B/n_B, I_A)
  \]

- **KTS 11**: Basic Kerberos authentication protocol [11]
  
  \[
  A \rightarrow T : r_A, I_A, I_B \\
  A \leftarrow T : E_{K_{AT}}(r_A, K, I_B), E_{K_{BT}}(r_A, K, I_A) \\
  A \rightarrow B : E_{K_{BT}}(r_A, K, I_A) \\
  A \leftarrow B : E_{K}(r_A, K)
  \]

3. Five-Pass

- **KTS 13**: Mechanism 9 of [15]
  
  \[
  A \leftarrow B : r_B \\
  A \rightarrow T : r_A, r_B, I_B \\
  A \leftarrow T : E_{K_{AT}}(r_A, K, I_B), E_{K_{BT}}(r_B, K, I_A) \\
  A \rightarrow B : E_{K_{BT}}(r_B, K, I_A), E_{K}(r'_A, r'_B) \\
  A \leftarrow B : E_{K}(r_B, r'_A)
  \]

- **KTS 14**: Needham-Schroeder shared-key protocol [11]
  
  \[
  A \rightarrow T : I_A, I_B, r_A \\
  A \leftarrow T : E_{K_{AT}}(r_A, I_B, K, E_{K_{BT}}(r_B, K, I_A)) \\
  A \rightarrow B : E_{K_{BT}}(K, I_A) \\
  A \leftarrow B : E_{K}(r_B) \\
  A \leftarrow B : E_{K}(r_B - 1)
  \]

3.1.3 KTC

1. Three-Pass

- **KTS 15**: Mechanism 11 of [15]
  
  \[
  A \rightarrow T : E_{K_{AT}}(I_B, K) \\
  A \leftarrow T : E_{K_{BT}}(K, I_A) \\
  A \leftarrow B : E_{K_{BT}}(K, I_A)
  \]
2. Four-Pass

- KTS 16: Mechanism 12 of [15]
  \[ \text{A} \rightarrow \text{T} : E_{K_{AT}}(TV_{PA}, IB, K) \]
  \[ \text{A} \rightarrow \text{T} : E_{K_{AT}}(TV_{PA}, IB), E_{K_{BT}}(t_{T}/n_{T}, K, I_{A}) \]
  \[ \text{A} \rightarrow \text{B} : E_{K_{BT}}(t_{T}/n_{T}, K, I_{A}), E_{K}(t_{A}/n_{A}, IB) \]
  \[ \text{A} \rightarrow \text{B} : E_{K}(t_{B}/n_{B}, IB) \]

3. Five-Pass

- KTS 17: Mechanism 13 of [15]
  \[ \text{A} \leftarrow \text{B} : r_{B} \]
  \[ \text{A} \rightarrow \text{T} : P_{B}(K_{A}, I_{A}) \]
  \[ \text{A} \rightarrow \text{B} : P_{B}(K_{B}) \]
  \[ K = f(K_{A}, K_{B}) \]

3.2 Asymmetric techniques

This section discusses key transport protocols based on asymmetric techniques. They are subdivided into protocols with and without signature.

3.3 Key Transport without signature

1. One-Pass

- KTA 1: Key transport mechanism 1 of [16]
  \[ \text{A} \rightarrow \text{B} : P_{B}(I_{A}, K, TVP) \]
  \[ \text{ex) ElGamal, RSA} \]

2. Three-Pass

  \[ \text{A} \rightarrow \text{B} : P_{B}(K_{A}, I_{A}) \]
  \[ \text{A} \rightarrow \text{B} : P_{A}(K_{A}, K_{B}) \]
  \[ \text{A} \rightarrow \text{B} : P_{B}(K_{B}) \]
  \[ K = f(K_{A}, K_{B}) \]

- KTA 3: Key transport mechanism 6 of [16]
  \[ \text{A} \rightarrow \text{B} : P_{B}(I_{A}, K_{A}, r_{A}) \]
  \[ \text{A} \rightarrow \text{B} : P_{A}(I_{B}, K_{B}, r_{A}, r_{B}) \]
  \[ \text{A} \rightarrow \text{B} : r_{B} \]
  \[ K = f(K_{A}, K_{B}) \]

3.4 Key Transport with signature

1. One-Pass

  \[ \text{A} \rightarrow \text{B} : P_{B}(K, t_{A}, S_{A}(I_{B}, K, t_{A})) \]

  \[ \text{A} \rightarrow \text{B} : P_{B}(K, t_{A}), S_{A}(I_{B}, K, t_{A}) \]

  \[ \text{A} \rightarrow \text{B} : t_{A}, P_{B}(I_{B}, K), S_{A}(I_{B}, t_{A}, P_{B}(I_{B}, K)) \]
2. Two-Pass

- **KTA 7:** Key transport mechanism 4 of [16]
  - $A \rightarrow B : r_A$
  - $A \leftarrow B : I_A, r_A, r_B, P_A(I_B, K), S_B(I_A, r_A, r_B, P_A(I_B, K))$

- **KTA 8:** X.509 strong two-way authentication [11]
  - $A \rightarrow B : t_A, r_A, I_B, P_B(K_A)$
  - $A \leftarrow B : t_B, r_B, I_A, r_A, P_A(K_B)$

3. Three-Pass

- **KTA 9:** Key transport mechanism 5 of [16]
  - $A \rightarrow B : r_A$
  - $A \leftarrow B : S_B(r_A, r_B, I_B, P_A(I_B, K_B))$
  - $A \rightarrow B : S_A(r_A, r_B, I_B, P_B(I_A, K_A))$

- **KTA 10:** X.509 strong three-way authentication [11]
  - $A \rightarrow B : r_A, I_B, P_B(K_A), S_A(r_A, I_B, P_B(K_A))$
  - $A \leftarrow B : r_B, I_A, r_A, P_A(K_B), S_B(r_B, I_A, r_A, P_A(K_B))$
  - $A \rightarrow B : r_B, I_B, S_A(r_B, I_B)$

3.5 Hybrid Key Transport

Hybrid key transport protocol uses symmetric encryption in addition to both asymmetric encryption and signatures.

1. Two-Pass

- **KTA 11:** 2-pass Beller-Yacobi key transport [11]
  - $A \leftarrow B : r_B, Cert_B$
  - $A \rightarrow B : P_B(v), E_{v}(Cert_A, w)$ (where $(v, w) = S_A(r_B, I_B)$)

2. Four-Pass

- **KTA 12:** 4-pass Beller-Yacobi key transport [11]
  - $A \leftarrow B : Cert_B$
  - $A \rightarrow B : P_B(K)$
  - $A \leftarrow B : E_K(m)$
  - $A \rightarrow B : E_K(S_A(m), Cert_A)$

4 Key Agreement

Key agreement protocols can be classified into protocols using symmetric and asymmetric encipherment techniques.

4.1 Symmetric techniques

This section concisely presents key agreement protocols based on symmetric techniques. The protocols are grouped in accordance with the use of a TTP.
4.1.1 Point-to-Point

1. Two-Pass

- **KAS 1**: Mechanism 5 of [15]
  \[ \begin{align*}
  & A \rightarrow B : E_{K_{AB}}(t_A/n_A, I_B, K_A) \\
  & A \leftarrow B : E_{K_{AB}}(t_B/n_B, I_A, K_B) \\
  & K = f(K_A, K_B)
  \end{align*} \]

2. Three-Pass

- **KAS 2**: Mechanism 6 of [15]
  \[ \begin{align*}
  & A \leftarrow B : r_B \\
  & A \rightarrow B : E_{K_{AB}}(r_A, r_B, I_B, K_A) \\
  & A \leftarrow B : E_{K_{AB}}(r_B, r_A, K_B) \\
  & K = f(K_A, K_B)
  \end{align*} \]

4.1.2 TTP (CA)

- **KAS 3**: Blom’s scheme [11]
  
  (public information) \( p_A, p_B \in \mathbb{Z}_p \)
  
  \[ \begin{align*}
  & T : a, b, c \in \mathbb{Z}_p \\
  & f(x, y) = a + b(x + y) + cxy \mod p \\
  & g_A(x) = f(x, p_A) \\
  & g_B(x) = f(x, p_B) \\
  & T \Rightarrow A : g_A(x) \\
  & T \Rightarrow B : g_B(x) \\
  & A : K = g_A(p_B), \ B : K = g_B(p_A) \\
  & K = f(p_A, p_B)
  \end{align*} \]

4.2 Asymmetric techniques

This section describes the briefs of key agreement protocols using asymmetric techniques. They are based on Diffie-Hellman protocol[6].

4.2.1 Diffie-Hellman and related key agreement protocols

1. Non-interactive key agreement protocol

- **KAA 1**: Static Diffie-Hellman key agreement [11]
  
  \[ \begin{align*}
  & A \cdots B \\
  & A : K = z_A^b \mod p, \ B : K = z_A^b \mod p \\
  & K = g^{ab} \mod p
  \end{align*} \]

2. One-Pass

- **KAA 2**: ElGamal key agreement [11]
  
  \[ \begin{align*}
  & A \rightarrow B : g^a \mod p \\
  & A : K = z_B^a \mod p, \ B : K = (g^a)^b \mod p \\
  & K = g^{ab} \mod p
  \end{align*} \]

- **KAA 3**: Nyberg-Rueppel key agreement [15]
  
  \[ \begin{align*}
  & A : e = g^e \mod p \\
  & K = z_B^r \mod p \\
  & e' = e \cdot H(K, I_A, TVP) \mod p \\
  & y = r - ae' \\
  & A \rightarrow B : I_A, TVP, e, y
  \end{align*} \]
3. Two-Pass

- **KAA 4:** Ephemeral Diffie-Hellman key agreement [6]
  A $\rightarrow$ B : $g^x \mod p$
  A $\leftarrow$ B : $g^y \mod p$
  A : $K = (g^y)^x \mod p$, B : $K = (g^x)^y \mod p$
  $K = g^{xy} \mod p$

- **Matsumoto-Takashima-Imai (MTI) key agreement [10]**
  - **KAA 5-1:** MTI A(0) key agreement
    A $\rightarrow$ B : $z_A^x \mod p$
    A $\leftarrow$ B : $z_A^y \mod p$
    A : $K = (z_A^y)^{-1}g^x \mod p$, B : $K = (z_A^x)^{-1}g^y \mod p$
    $K = g^{xy} \mod p$
  - **KAA 5-2:** MTI B(0) key agreement
    A $\rightarrow$ B : $z_B^x \mod p$
    A $\leftarrow$ B : $z_B^y \mod p$
    A : $K = (z_B^y)^{-1}g^x \mod p$, B : $K = (z_B^x)^{-1}g^y \mod p$
    $K = g^{xy} \mod p$
  - **KAA 5-3:** MTI C(0) key agreement
    A $\rightarrow$ B : $z_B^{ax} \mod p$
    A $\leftarrow$ B : $z_B^{ay} \mod p$
    A : $K = (z_B^{ay})^x \mod p$, B : $K = (z_B^{ax})^y \mod p$
    $K = g^{xay} \mod p$
  - **KAA 5-4:** MTI C(1) key agreement
    A $\rightarrow$ B : $z_B^{ax} \mod p$
    A $\leftarrow$ B : $z_B^{ay} \mod p$
    A : $K = (z_B^{ay})^x \mod p$, B : $K = (z_B^{ax})^y \mod p$
    $K = g^{xay} \mod p$

- **KAA 6:** LLK key agreement [9]
  A $\rightarrow$ B : $z_B^x \mod p$
  A $\leftarrow$ B : $z_B^y \cdot (z_B^x)^{y/b} \mod p$
  A : $K = z_B^x \cdot (z_B^y)^{(a+x)} \mod p$, B : $K = z_B^y \cdot z_A^x \mod p$
  $K = g^{xy+bz} \mod p$

- **KAA 7:** KEA key agreement [13]
  A $\rightarrow$ B : $g^x \mod p$
  A $\leftarrow$ B : $g^y \mod p$
  A : $K = (g^y)^x + z_A^y \mod p$, B : $K = (g^x)^y + z_B^y \mod p$
  $K = g^{xy} + g^{by} \mod p$

- **KAA 8:** Unified Model key agreement [1]
  A $\rightarrow$ B : $g^x \mod p$
  A $\leftarrow$ B : $g^y \mod p$
  A : $K = H(z_B^a, (g^y)^x)$, B : $K = H(z_B^b, (g^y)^y)$
  $K = K = H(g^{ob}, g^{xy})$

- **KAA 9:** MQV key agreement [8]
  A $\rightarrow$ B : $g^x \mod p$
  A $\leftarrow$ B : $g^y \mod p$
  A : $s_A = (x + ag^x) \mod q$, $K = (g^y(z_B)\overline{g})^{s_A} \mod p$, B : $s_B = (y + bg^y) \mod q$, $K = (g^x(z_A)\overline{g})^{s_B} \mod p$
  $K = g^{s_A+s_B} \mod p$
KAA 10: Song-Kim key agreement [12]

\[ A \rightarrow B : g^x \mod p \]
\[ A \leftarrow B : g^y \mod p \]
\[ A : K = (g^y)^{(a+x) \cdot z} \mod p, \quad B : K = (g^x)^{(b+y) \cdot z} \mod p \]
\[ K = g^{xy+ay+bx} \mod p \]

4. Three-Pass

KAA 11: Station-to-Station protocol (STS) [5]

\[ A \rightarrow B : g^x \mod p \]
\[ A \leftarrow B : g^y \mod p, E_K(S_B(g^y, g^x, I_A)) \]
\[ A \rightarrow B : E_K(S_A(g^x, g^y, I_B)) \]
\[ A : K = (g^y)^x \mod p, \quad B : K = (g^x)^y \mod p \]
\[ K = g^{xy} \mod p \]

KAA 12: AKC protocols [1]

\[ A \rightarrow B : g^x \mod p \]
\[ A \leftarrow B : g^y \mod p, MAC_K(2, g^y, g^x, I_B, I_A) \]
\[ A \rightarrow B : MAC_K(3, g^x, g^y, I_A, I_B) \]

ex) MTI AKC, KEA AKC, Unified Model AKC, MQV AKC, Song-Kim AKC

5 Comparison

This section compares key establishment protocols classified in Sections 3 and 4. Table 1 and Table 2 summarize the properties of the key transport protocols using symmetric techniques and asymmetric techniques, respectively. In Table 3 and Table 4, the major properties of the key agreement protocols using symmetric techniques and asymmetric techniques are showed, respectively.

[ Symbol ]

A The property is provided to A in the protocol.
A, B The property is provided to both entities, A and B in the protocol.
A \rightarrow B The property is provided from A to B in the protocol.
A \leftarrow B The property is provided from A to B and from B to A.
(A) The property can be provided to A as an option in the protocol by using additional means.
. The property is not provided in the protocol.
◊ The property is provided in the protocol.
t/n, r The property is provided by using t/n, r.
i, j The number of computations of asymmetric transformation.
(i.e., A needs i computations and B needs j computations.)

[ Abbreviation ]

Sig Signature
TTP Trust Third Party
NP Number of Passes
KA Key Authentication (key integrity and key origin authentication)
IKA Implicit Key Authentication
EKA Explicit Key Authentication
Con Key Confirmation
EA Entity Authentication
PKO Public Key Operation
KF Key Freshness
KC Key Control
KTS Key Transport using Symmetric techniques
KTA Key Transport using Asymmetric techniques
KAS Key Agreement using Symmetric techniques
KAA Key Agreement using Asymmetric techniques
Table 1: Comparison of Key Transport Protocols using Symmetric Techniques

<table>
<thead>
<tr>
<th>KTS</th>
<th>TTP</th>
<th>NP</th>
<th>KA</th>
<th>Con</th>
<th>EA</th>
<th>KC</th>
<th>KF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>A</td>
<td>·</td>
<td>·</td>
</tr>
<tr>
<td>3</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>A</td>
<td>A</td>
<td>T/n</td>
<td>·</td>
</tr>
<tr>
<td>4</td>
<td>·</td>
<td>2</td>
<td>·</td>
<td>·</td>
<td>A</td>
<td>A</td>
<td>r</td>
</tr>
<tr>
<td>5</td>
<td>·</td>
<td>3</td>
<td>o</td>
<td>A,B</td>
<td>B</td>
<td>r</td>
<td>·</td>
</tr>
<tr>
<td>6</td>
<td>·</td>
<td>3</td>
<td>o</td>
<td>o</td>
<td>A,B</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>7</td>
<td>·</td>
<td>3</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>A</td>
</tr>
<tr>
<td>8</td>
<td>KDC</td>
<td>3</td>
<td>o</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>KDC</td>
</tr>
<tr>
<td>9</td>
<td>KDC</td>
<td>3</td>
<td>o</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>KDC</td>
</tr>
<tr>
<td>10</td>
<td>KDC</td>
<td>4</td>
<td>o</td>
<td>o</td>
<td>A,B</td>
<td>KDC</td>
<td>t/n</td>
</tr>
<tr>
<td>11</td>
<td>KDC</td>
<td>4</td>
<td>o</td>
<td>o</td>
<td>A,(B)</td>
<td>KDC</td>
<td>t/n</td>
</tr>
<tr>
<td>12</td>
<td>KDC</td>
<td>4</td>
<td>o</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>KDC</td>
</tr>
<tr>
<td>13</td>
<td>KDC</td>
<td>5</td>
<td>o</td>
<td>o</td>
<td>A,B</td>
<td>KDC</td>
<td>r</td>
</tr>
<tr>
<td>14</td>
<td>KDC</td>
<td>5</td>
<td>o</td>
<td>o</td>
<td>A</td>
<td>KDC</td>
<td>r</td>
</tr>
<tr>
<td>15</td>
<td>KTC</td>
<td>3</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>·</td>
</tr>
<tr>
<td>16</td>
<td>KTC</td>
<td>4</td>
<td>o</td>
<td>o</td>
<td>A,B</td>
<td>A</td>
<td>t/n</td>
</tr>
<tr>
<td>17</td>
<td>KTC</td>
<td>5</td>
<td>o</td>
<td>o</td>
<td>A,B</td>
<td>B</td>
<td>r</td>
</tr>
</tbody>
</table>

Table 2: Comparison of Key Transport Protocols using Asymmetric Techniques

<table>
<thead>
<tr>
<th>KTA</th>
<th>Sig</th>
<th>NP</th>
<th>IKA</th>
<th>Con</th>
<th>EA</th>
<th>KC</th>
<th>KF</th>
<th>PKO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>·</td>
<td>1</td>
<td>A</td>
<td>B,</td>
<td>·</td>
<td>A</td>
<td>A</td>
<td>TVP</td>
</tr>
<tr>
<td>2</td>
<td>·</td>
<td>3</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>3,3</td>
</tr>
<tr>
<td>3</td>
<td>·</td>
<td>3</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>4</td>
<td>o</td>
<td>1</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>t</td>
</tr>
<tr>
<td>5</td>
<td>o</td>
<td>1</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>t</td>
</tr>
<tr>
<td>6</td>
<td>o</td>
<td>1</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>t</td>
</tr>
<tr>
<td>7</td>
<td>o</td>
<td>2</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>8</td>
<td>o</td>
<td>2</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>9</td>
<td>o</td>
<td>3</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>10</td>
<td>o</td>
<td>3</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>11</td>
<td>o</td>
<td>2</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>r</td>
</tr>
<tr>
<td>12</td>
<td>o</td>
<td>4</td>
<td>A</td>
<td>B,</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>r</td>
</tr>
</tbody>
</table>

Table 3: Comparison of Key Agreement Protocols using Symmetric Techniques

<table>
<thead>
<tr>
<th>KAS</th>
<th>TTP</th>
<th>NP</th>
<th>KA</th>
<th>Con</th>
<th>EA</th>
<th>KF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>·</td>
<td>2</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>t/n</td>
</tr>
<tr>
<td>2</td>
<td>·</td>
<td>3</td>
<td>o</td>
<td>·</td>
<td>·</td>
<td>r</td>
</tr>
<tr>
<td>3</td>
<td>o</td>
<td>1</td>
<td>o</td>
<td>·</td>
<td>·</td>
<td>·</td>
</tr>
</tbody>
</table>
Table 4: Comparison of Key Agreement Protocols using Asymmetric Techniques

<table>
<thead>
<tr>
<th>KAA</th>
<th>Sig</th>
<th>NP</th>
<th>IKA</th>
<th>Con</th>
<th>EA</th>
<th>KF</th>
<th>PKO</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>·</td>
<td>0</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>1,1</td>
</tr>
<tr>
<td>2</td>
<td>·</td>
<td>1</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>2,1</td>
</tr>
<tr>
<td>3</td>
<td>o</td>
<td>1</td>
<td>A ↔ B</td>
<td>A ↔ B</td>
<td>A</td>
<td>o</td>
<td>2,3</td>
</tr>
<tr>
<td>4</td>
<td>·</td>
<td>2</td>
<td>·</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>2,2</td>
</tr>
<tr>
<td>5-1</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>3,3</td>
</tr>
<tr>
<td>5-2</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>3,3</td>
</tr>
<tr>
<td>5-3</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>2,2</td>
</tr>
<tr>
<td>5-4</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>2,2</td>
</tr>
<tr>
<td>6</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>2,2</td>
</tr>
<tr>
<td>7</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>3,3</td>
</tr>
<tr>
<td>8</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>3,3</td>
</tr>
<tr>
<td>9</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>2.5,2.5</td>
</tr>
<tr>
<td>10</td>
<td>·</td>
<td>2</td>
<td>A ↔ B</td>
<td>·</td>
<td>·</td>
<td>o</td>
<td>3,3</td>
</tr>
<tr>
<td>11</td>
<td>o</td>
<td>3</td>
<td>A ↔ B</td>
<td>A ↔ B</td>
<td>A,B</td>
<td>o</td>
<td>3,3</td>
</tr>
<tr>
<td>12</td>
<td>·</td>
<td>3</td>
<td>A ↔ B</td>
<td>A ↔ B</td>
<td>A,B</td>
<td>o</td>
<td>3,3</td>
</tr>
</tbody>
</table>

All the key establishment protocols using symmetric techniques offer at least implicit key authentication, because only parties knowing a specific secret key can recover the correct key. Key confirmation and entity authentication can be achieved for every protocol by using the some techniques[15]. Distinguishing identifiers included in messages are used to protect against certain type of substitution attacks.

6 Concluding Remarks

We have surveyed and classified many key establishment protocols proposed so far in accordance with cryptographic techniques, models of environment used, and the number of pass. And then the major properties of the protocols has been compared. Most important properties of key establishment protocols depend on the structure of the messages exchanged, not on the underlying cryptographic algorithms. Therefore, for reducing errors in key establishment protocol, the prudent design of protocol messages is rather of consequence. One of the important tasks of key establishment protocol is to devise protocols providing formally provable security[2, 3].

References


