Cryptography

Cunsheng DING, HKUST

COMP3631

Lecture 13: Two Mutual Authentication Protocols

Topics of This Lecture

1. Passive and active attacks.
2. A data origin authentication protocol.
3. A sender authentication protocol.
4. A mutual authentication protocol based on a preshared key.
5. A challenge-response mutual authentication protocol.

Part I: Passive and Active Attacks

Passive and Active Attacks

Passive attacks: Any attack on a security system under the assumption that the attacker can only intercept texts exchanged over a communication channel is called a **passive attack**.

Active attacks: Any attack on a security system under the assumption that the attacker can stop, intercept, delete, modify, and replay texts exchanged over a communication channel or insert his/her texts into the channel is called an **active attack**. In such a scenario, we say that the attacker has **full control** over the communication channel.

Remark: Active attacks are more powerful than passive attacks.

Remark: A system may be secure with respect to passive attacks, but insecure with respect to active attacks. See Lecture 14 for such a system.

Part II: A Data Origin Authentication Protocol

- Recall the **data origin authentication** security service defined in Lecture 3. It means the verification of the identity of the **creator** of a piece of data or a message.
- In this part, we introduce the first data origin authentication protocol for providing this security service.

Identifiers (IDs) of People and Objects

Definition: They are strings of characters for identifying people or objects.

Example: HK ID numbers, HKUST student numbers, HKUST staff numbers, product serial numbers

Remarks: An ID has the following properties:

- It is a constant and used for identifying only one person or object.
- It is generated and issued by an organisation or an issuer.
- It is not a secret.
- It has a specified format and fixed length in a specific protocol.

ID database: For certain applications, the issuer maintains a public directory of all IDs, which is called the ID database.

A Data Origin Authentication Protocol

The protocol: Alice and Bob share a secret key k for a secure one-key cipher.

$$\text{Alice } \rightarrow E_k(ID_A||ID_B)||ID_A||ID_B \rightarrow \text{ Bob},$$

where ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

The format of the message above is:

Alice $\rightarrow$ autentication value||alledged sender name||receiver name $\rightarrow$ Bob.

Question: Who can create this message $E_k(ID_A||ID_B)||ID_A||ID_B$?

Question: How should Bob process the received text?

Question: What security services can this protocol provide?

A Data Origin Authentication Protocol

The protocol: Alice and Bob share a secret key k for a secure one-key cipher.

$$\text{Alice} \rightarrow E_k(ID_A||ID_B)||ID_A||ID_B \rightarrow \text{Bob},$$

where ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

- It provides the data integrity service. Why?
- It provides the data origin authentication service. Why?
- It does not provide the sender authentication service with respect to active attacks, as it may be a replayed earlier message.

Question: How do you modify it into a sender authentication protocol?

Part III: A Sender Authentication Protocol

- Recall the **sender authentication** security service defined in Lecture 3. It means the verification of the identity of the sender of a piece of received data.
- Recall that the sender of a piece of data may not be the creator of the data, as the data may be a replayed earlier message. Hence, the sender authentication is different from the data origin authentication.
- In this part, we introduce the first sender authentication protocol, which is a modified version of the previous data origin authentication protocol.

The Sender Authentication Protocol

The protocol: Alice and Bob share a secret key k for a secure one-key cipher.

$$\text{Alice } \rightarrow E_k(Timestamp||ID_A||ID_B)||ID_A||ID_B \rightarrow \text{ Bob},$$

where $Timestamp$ is the current time, ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

Question: How should Bob process the received text?

Answer: This is similar to the previous protocol.

Question: What security services can this protocol provide?

The Sender Authentication Protocol

The protocol: Alice and Bob share a secret key k for a secure one-key cipher.

$$\text{Alice } \rightarrow E_k(Timestamp||ID_A||ID_B)||ID_A||ID_B \rightarrow \text{ Bob},$$

where $Timestamp$ is the current time, ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

- It provides the data integrity service. Why?
- It provides the antireplay service due to the timestamp.
- It provides the data origin authentication and the sender authentication service. Why?

Part IV: Two Mutual Authentication Protocols

- By the **mutual authentication** security service we mean the verification of the identities of two parties who are communicating with each other.
- In this part, we introduce two mutual authentication protocols for providing this security service.

The First Mutual Authentication Protocol

The protocol: Alice and Bob share a secret key k for a secure one-key cipher.

$$\text{Alice} \rightarrow E_k(Timestamp_1||ID_A||ID_B)||ID_A||ID_B \rightarrow \text{Bob},$$

$$\text{Alice} \leftarrow ID_A||ID_B||E_k(ID_A||ID_B||Timestamp_2) \leftarrow \text{Bob},$$

where $Timestamp$ is the current time, ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

- It provides the data integrity service.
- It provides the antireplay service due to the timestamp.
- It provides the data origin authentication and the mutual authentication service.

The Concept of Nonces Used in Security Protocols

Definition: It is a one-time variable string of characters.

Example: A timestamp, a sequence number, a random string.

Remarks: A nonce has the following properties:

- It is variable (it is not a constant).
- It is used only once for the same purpose in a protocol.
- It has a specified format and fixed length in a specific protocol.

The Second Mutual Authentication Protocol

The challenge-response protocol: Alice and Bob have exchanged their public keys for a secure public-key cipher.

$$\text{Alice } \rightarrow E_{k_e^{(B)}}(N_1)||ID_A||ID_B \rightarrow \text{ Bob},$$

$$\text{Alice } \leftarrow ID_A||ID_B||E_{k_e^{(A)}}(N_1||N_2) \leftarrow \text{ Bob},$$

$$\text{Alice } \rightarrow E_{k_e^{(B)}}(N_2)||ID_A||ID_B \rightarrow \text{ Bob},$$

where N_i are nonces, ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

Question: How should these received texts be processed by their receivers?

Conclusions: It provides the data confidentiality, the data integrity, the antireplay and the mutual authentication services. Why?

Part V: Remarks on the Authentication Protocols

- The two mutual authentication protocols are widely used in real-world security systems!!
- The first mutual authentication protocol is based on a preshared secret key, while the second does not use a preshared key.
- Other data origin authentication protocols will be introduced later.
- Notice that both symmetric and asymmetric encryption may provide security services other than the data confidentiality service.

  – This is a partial answer to the question on Slide Number 21 of Lecture 9.

Online Demo of Some One-Key Ciphers

Please use the online demo below to understand why some of these protocols could provide the claimed security services:

`https://8gwifi.org/CipherFunctions.jsp`

Appendix: Some Applications of the Protocols

Question: How do you combine a mutual authentication protocol in this lecture with other applications?

An Application of a Protocol

The application: Alice and Bob share a secret key k for a secure one-key cipher, and would exchange their texts $text_1$ and $text_2$.

$$\text{Alice} \rightarrow E_k(text_1||Timestamp_1||ID_A||ID_B)||ID_A||ID_B \rightarrow \text{Bob},$$

$$\text{Alice} \leftarrow ID_A||ID_B||E_k(ID_A||ID_B||Timestamp_2||text_2) \leftarrow \text{Bob},$$

where $Timestamp$ is the current time, ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

- It provides the data confidentiality, data integrity, antireplay, the data origin authentication and the mutual authentication services.
- One of the texts $text_1$ and $text_2$ could be empty.

An Application of Another Protocol

The protocol: Alice and Bob have exchanged their public keys for a secure public-key cipher and would exchange their texts $text_1$ and $text_2$.

$$\text{Alice } \rightarrow E_{k_e^{(B)}}(text_1||N_1)||ID_A||ID_B \rightarrow \text{ Bob},$$

$$\text{Alice } \leftarrow ID_A||ID_B||E_{k_e^{(A)}}(N_1||N_2||text_2) \leftarrow \text{ Bob},$$

$$\text{Alice } \rightarrow E_{k_e^{(B)}}(N_2)||ID_A||ID_B \rightarrow \text{ Bob},$$

where N_i are nonces, ID_A and ID_B are the ID of Alice and Bob, respectively. They know the ID of each other.

Remark: One of the texts $text_1$ and $text_2$ could be empty.

Question: How should these received texts be processed by their receivers? What security services are provided?