2020/01/21 @ 상명대학교

Protocols for Authentication and Key Establishment

- A Tutorial Introduction to Authentication and Key Establishment -

발표자: 최 창 준(changjun@pel.smuc.ac.kr)

지도교수: 이 종 혁(jonghyouk@smu.ac.kr)

상명대학교 프로토콜공학연구실

목 차

Introduction

- Subject of cryptographic protocols for authentication and key management is likely to be bemused by the sheer variety of techniques and technical background required
- Even before this stage is reached a more fundamental question needs to be faced
  - “What are these protocols there for at all?”
    - To answer this question it is necessary to provide an understanding of what sets cryptographic protocols apart from other types of protocols
- It provides necessary background material for those readers who are not already familiar with the topic of cryptographic protocols

Building a Key Establishment Protocol

- This section explication an attempt to design one good protocol
- Before designing any protocol the communications architecture must be established
- Our Scenario has a set of users, any two of whom may wish to establish a new key for use in securing their subsequent communications through cryptography
  - Such a key is known as a session key
- Once an appropriate key has been established its use comes in protecting the real data to be communicated with whatever cryptographic mechanisms are chosen

Building a Key Establishment Protocol

- In order to achieve their aim the users interact with an entity called the server which will also engage in the protocol
- All users trust the server to execute the protocol faithfully and not to engage in any other activity that will deliberately compromise their security
  - Furthermore, the server is trusted to generate the new key and to do so in such a way that it is sufficiently random to prevent an attacker gaining any useful information about it

Building a Key Establishment Protocol

- Protocols involve three entities
  - These are two users whom we denote A and B and the trusted server S
  - The aim of the protocol is for A and B to establish a new secret key K_{AB}
  - The role of S is to generate K_{AB} and transport it to A and B

- Aims of the Protocol
  1. At the end of the protocol the value of K_{AB} should be known to both A and B, but to no other with the possible exception of S
  2. A and B should know that K_{AB} is newly generated

Building a Key Establishment Protocol

- Protocol to achieve transport of a new session key K_{AB}

1. User A contacts S by sending the identities of the two parties who are going to share the new session key
2. Trusted Server S returns the key K_{AB} to A
3. User A passes K_{AB} on to B

- Generally use two different formats for protocol descriptions

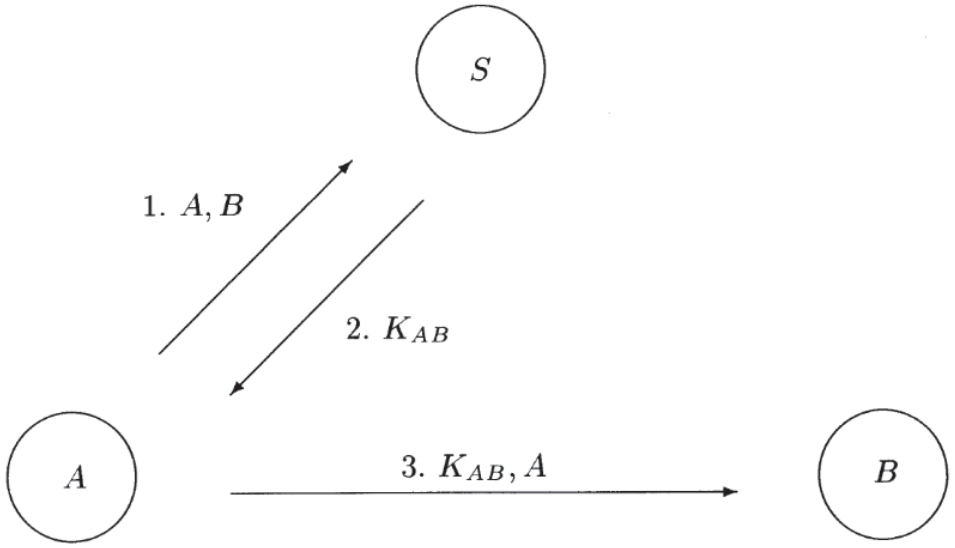

Fig. 1.1. First protocol attempt

1. $A \rightarrow S : A, B$
2. $S \rightarrow A : K_{AB}$
3. $A \rightarrow B : K_{AB}, A$

Protocol 1.1: First protocol attempt in conventional notation

Building a Key Establishment Protocol

- Confidentiality
  - **Security Assumption 1**
    - *The adversary is able to eavesdrop on all messages sent in a cryptographic protocol*
  - In order to provide confidentiality it is necessary to use a cryptographic algorithm and associated key
    - For now we will simply make the assumption that the server S initially shares a secret key with each user of the system
  - A passive eavesdropper cannot see K_{AB} since encrypted messages may only be read by the legitimate recipients who have the keys required to decrypt

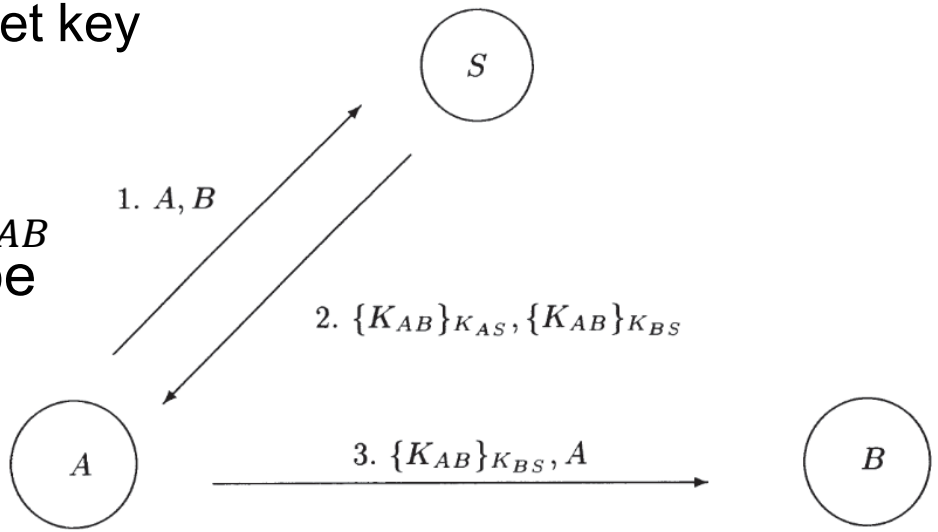

Fig. 1.2. Second protocol attempt

Building a Key Establishment Protocol

- Authentication
  - **Security Assumption 2**
    - *The adversary is able to alter all messages sent in a cryptographic protocol using any information available*
    - *In addition the adversary can reroute any message to any other principal*
      - *This includes the ability to generate and insert completely new messages*

  - The adversary C simply intercepts the message from A to B and substitutes D's identity for A's
    - Where D could be any identity including C's own

  - The consequence is that B believes that he is sharing the key with D whereas he is in fact sharing it with A

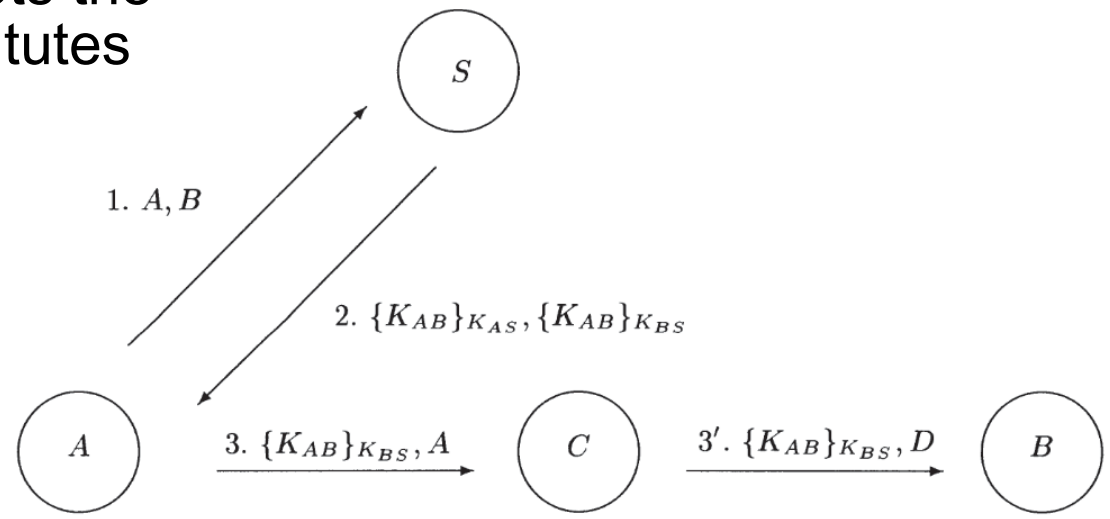

Fig. 1.3. Attack on the second protocol attempt

Building a Key Establishment Protocol

- Authentication
  - Another attack on the protocol does allow C to obtain the session key
    - C alters the message from A to S so that S encrypts the K_{AC} with C's key, K_{CS}, instead of with B's key
      - Since A cannot distinguish between encrypted messages meant for other principals she will not detect the alteration
  - Result of this attack
    - A will believe that the protocol has been successfully completed with B whereas in fact C knows K_{AC} and so can masquerade as B as well as learn all the information that A sends to B
  - In contrast to the previous attack
    - This one will only succeed if C is a legitimate user known to S

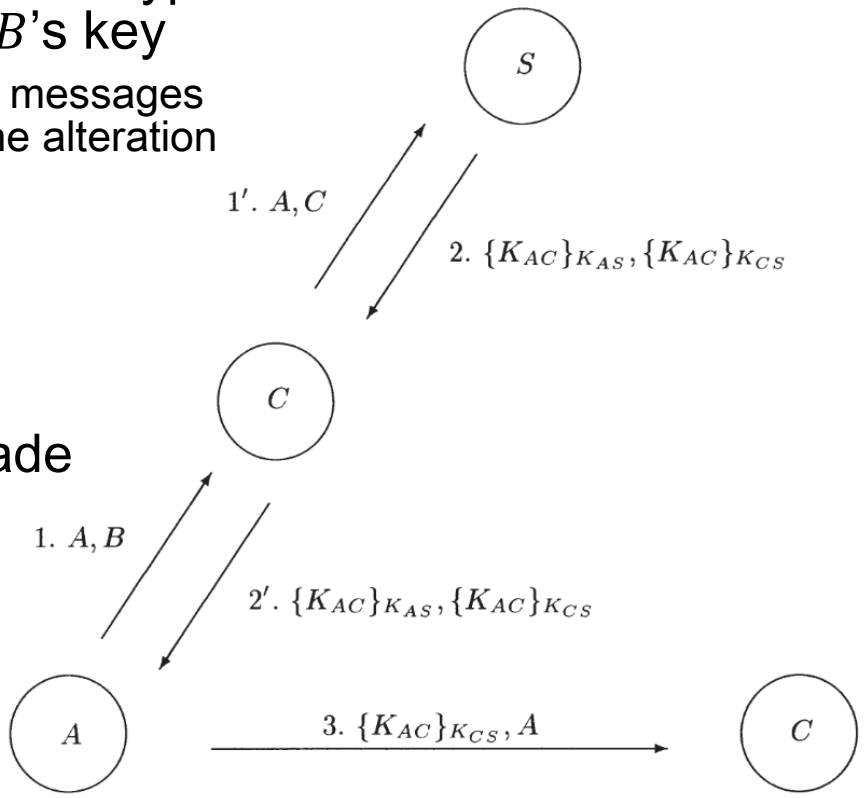

Fig. 1.4. Alternative attack on second protocol attempt

Building a Key Establishment Protocol

- Authentication
  - **Security Assumption 3**
    - *The adversary may be a legitimate protocol participant(an insider), or an external party(an outsider), or a combination of both*
  - To overcome the attack
    - The names of the users who are to share K_{AB} need to be bound cryptographically to the value of K_{AB}
    - Where the names of A and B are included in the encrypted messages received from S
    - It can easily be checked that in this protocol neither of the two attacks on the protocol of Fig. 1.2 will succeed

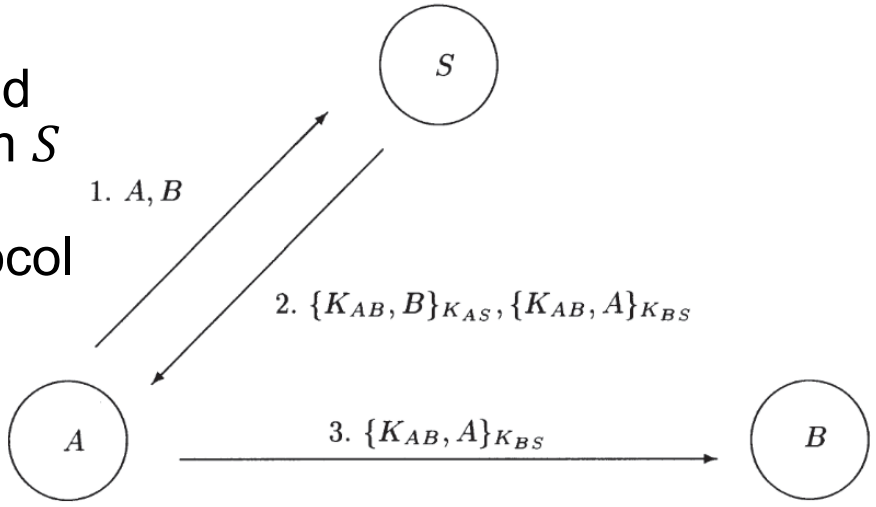

Fig. 1.5. Third protocol attempt

Building a Key Establishment Protocol

- Replay
  - Reason that a new key is generated for each session
    1. Session keys are expected to be vulnerable to attack
       - They may be placed in relatively insecure storage and could easily be discarded carelessly after the session is closed
    2. Communications in different sessions should be separated
       - In particular, it should not be possible to replay messages from previous sessions
  - For these reasons a whole class of attacks becomes possible based on the notion that old keys may be replayed in a subsequent session
  - Notice that even if A is careful in the management of session keys used by her, compromise of a session key by B may still allow replay attacks when A communicates with B

Building a Key Establishment Protocol

- Replay
  - **Security Assumption 4**
    - *An adversary is able to obtain the value of the session key K_{AB} used in any sufficiently old previous run of the protocol*
  - Replay attack on protocol
    - C intercepts the message from A to S
      - Indeed S plays no part in the protocol
    - The Key K'_{AB} is an old session key used by A and B in a previous session
    - By Security Assumption 1
      - C can be expected to know the encrypted messages via which K'_{AB} was transported to A and B
    - By Security Assumption 4
      - C can be expected to know the value of K'_{AB}

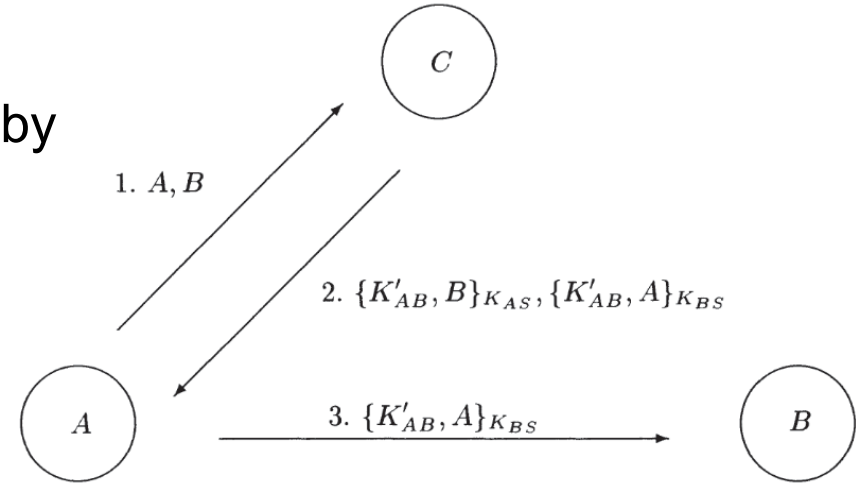

Fig. 1.6. Attack on third protocol attempt

Building a Key Establishment Protocol

- Replay
  - **Definition 1.1**
    - *A nonce is a random value generated by one party and returned to that party to show that a message is newly generated*
  - Generate nonce on protocol
    - User A send nonce N_A to S at the start of the protocol together with the request for a new key
      - If this same value is received with the session key then A can deduce that the key has not been replayed
    - Since B does not directly contact the S
      - It is inconvenient for him to send his own nonce to S to be returned with K_{AB}
    - User B generate a nonce N_B and send this to A protected by K_{AB} itself

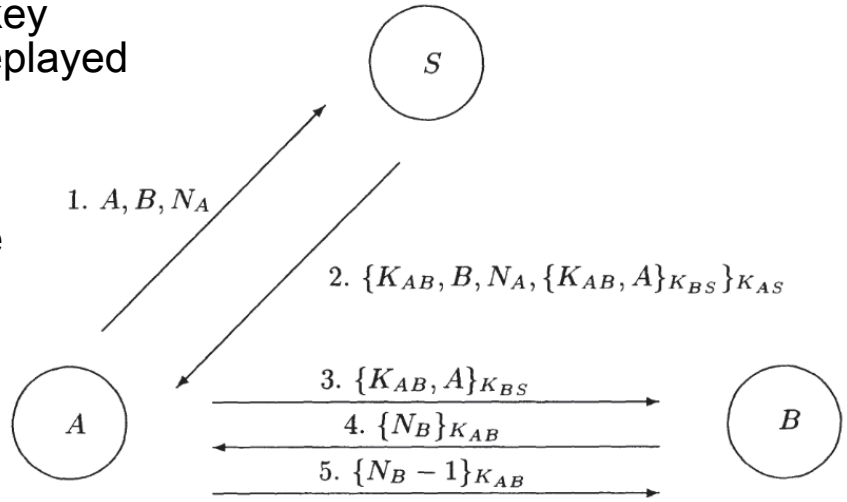

Fig. 1.7. Fourth protocol attempt (Needham–Schroeder)

Building a Key Establishment Protocol

- Replay
  - The protocol in Fig 1.7
    - Their attack illustrates that there was a flaw in the above argument used to justify the protocol design
      - This can be pinpointed to an assumption that only A will be able to form a correct reply to message 4 from B
  - In the attack in Fig 1.8
    - Since the adversary C can be expected to know the value of an old session key, this assumption is unrealistic
    - C masquerades as A and is thus able to persuade B to use the old key K'_{AB}

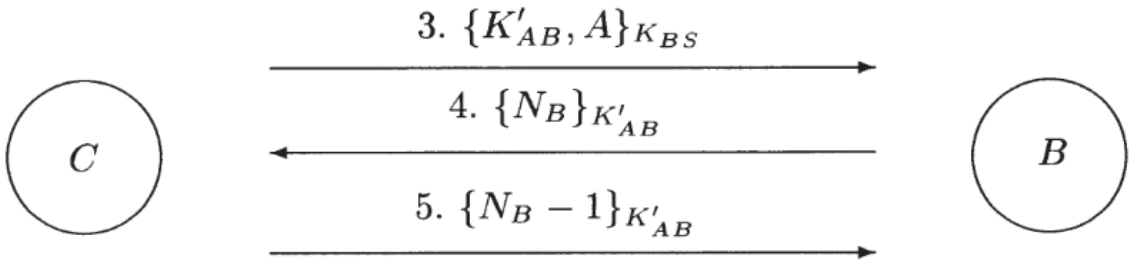

Fig. 1.8. Attack on fourth protocol attempt

Building a Key Establishment Protocol

- Replay
  - In the protocol of Fig 1.9
    - To enable both users to send their nonces to S
      - The protocol is now initiated by B who sends his nonce, N_B, first to A
    - A adds her nonce N_A, and sends both to S who is now able to return K_{AB} in separate messages for A and B
      - Which can each be verified as fresh by their respective recipients

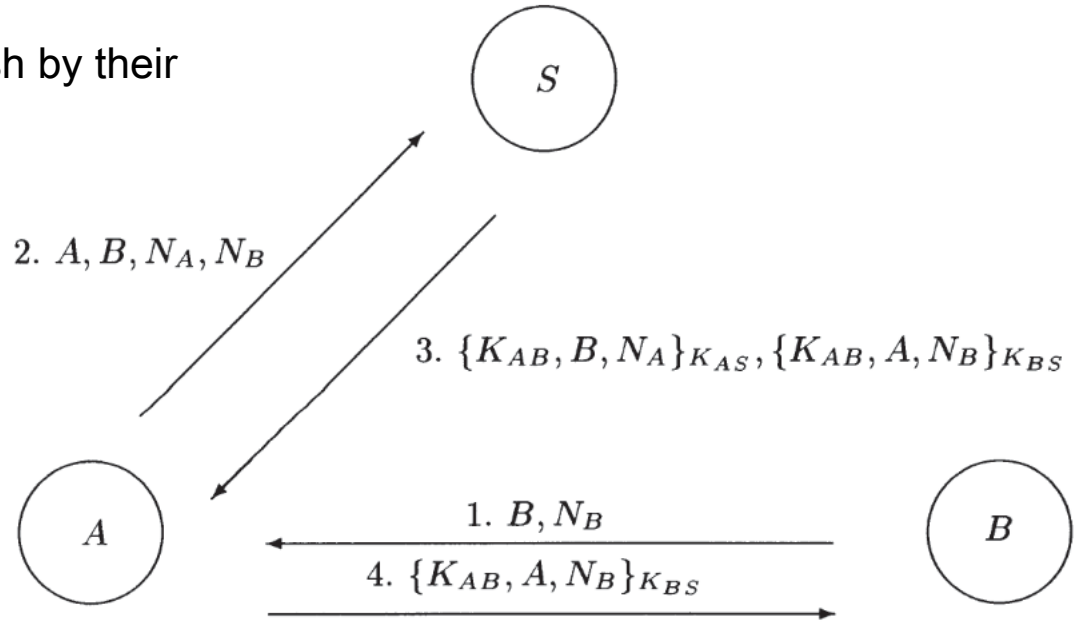

Fig. 1.9. Fifth protocol attempt

Building a Key Establishment Protocol

- It is worth nothing that it has been a very common pattern for published protocols to be subsequently found to be flawed
- Each time a new protocol is designed and an attack is found our understanding of protocol design improves
- The frequent occurrence of such attacks should be a caution, particularly for implementers of security protocols

Protocol Architecture

- TBA

Cryptographic Properties

- TBA

Freshness

- TBA

Types of Attack on Protocols

- TBA

Design Principles for Cryptographic Protocols

- TBA

Thanks!

최 창 준 (changjun@pel.smuc.ac.kr)