
The background of the slide is a light gray gradient, decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and subtle, scattered across the top and bottom edges of the frame.

# **REMOTE USER AUTHENTICATION USING SYMMETRIC ENCRYPTION**



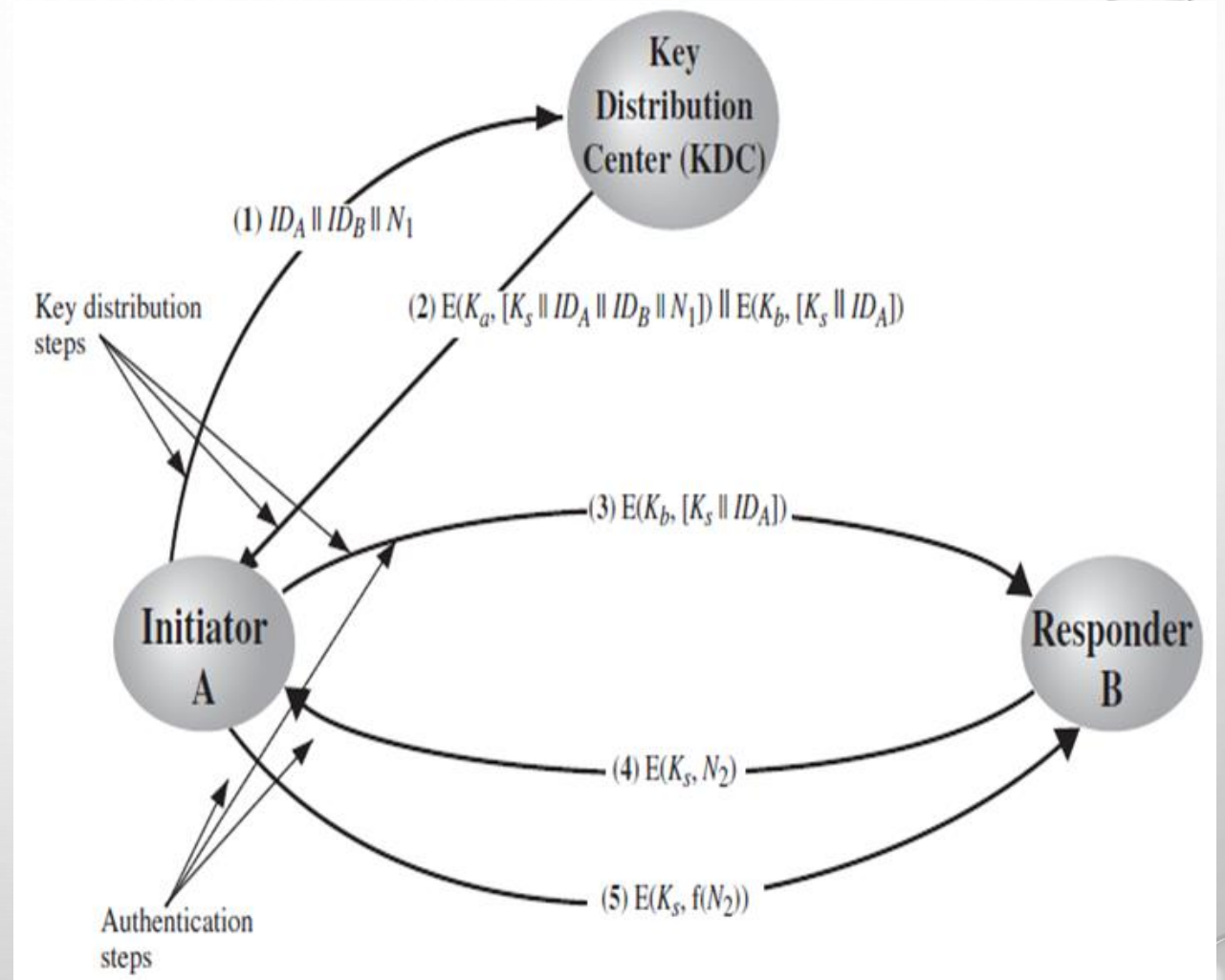
## **Introduction**

- Two-level hierarchy of symmetric encryption keys can be used to provide confidentiality for communication in a distributed environment.
  - In general, this strategy involves the use of a trusted Key Distribution Centre (KDC). Each party in the network shares a secret key, known as a master key, with the KDC.
  - The KDC is responsible for generating keys to be used for a short time over a connection (session key for logical connection) between two parties, known as session keys, and for distributing those keys using the master keys to protect the distribution.
- 

## Needham – Schroeder Protocol

The protocol can be summarized as follows.

$A \rightarrow KDC: ID_A || ID_B || N_1$   
 $KDC \rightarrow A: E(K_a, [K_s || ID_B || N_1 || E(K_b, [K_s || ID_A])])$   
 $A \rightarrow B: E(K_b, [K_s || ID_A])$   
 $B \rightarrow A: E(K_s, N_2)$   
 $A \rightarrow B: E(K_s, f(N_2))$



## **Disadvantages of Needham – Schroeder Protocol**

**The protocol is still vulnerable to a form of replay attack.**

Suppose that an opponent, X, has been able to compromise an old session key.

X can impersonate A and trick B into using the old key by simply replaying step 3.

Unless B remembers indefinitely all previous session keys used with A, B will be unable to determine that this is a replay. If X can intercept the handshake message in step 4, then it can impersonate A's response in step 5.

From this point on, X can send bogus messages to B that appear to B to come from A using an authenticated session key.

## Solution by Denning

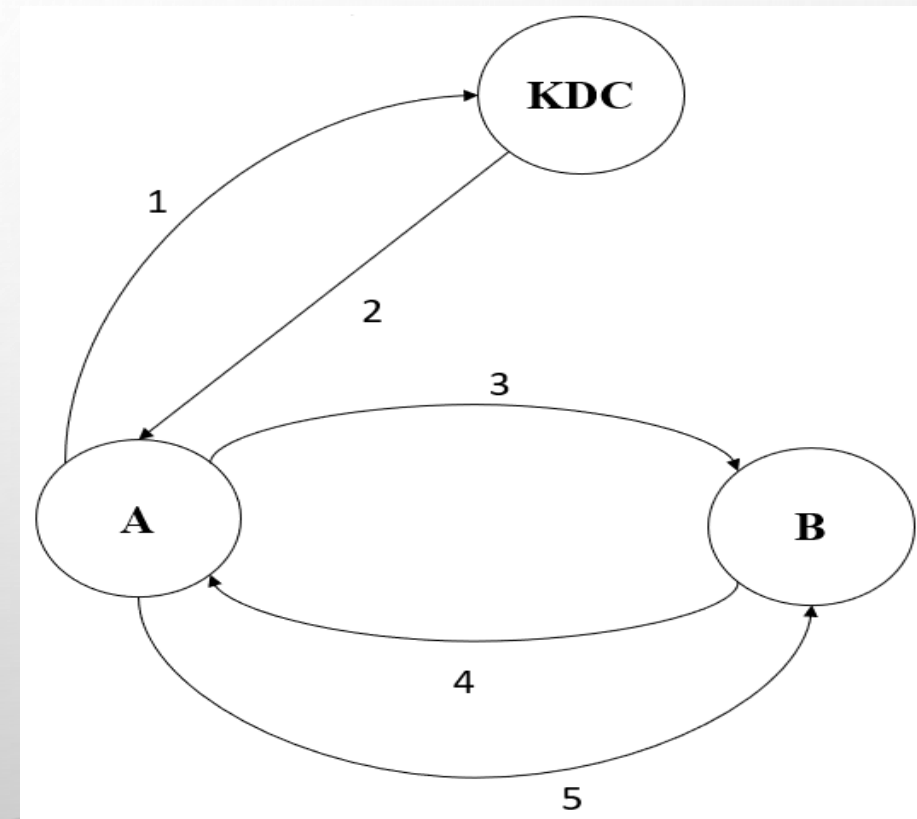
Denning proposes to overcome this weakness by a modification to the Needham/Schroeder protocol that includes the addition of a timestamp to steps 2 and 3.

Her proposal assumes that the master keys,  $K_a$  and  $K_b$ , are secure, and it consists of the following steps.

$A \rightarrow KDC: ID_A || ID_B$   
 $KDC \rightarrow A: E(K_a, [K_s || ID_B || T || E(K_b, [K_s || ID_A || T])])$   
 $A \rightarrow B: E(K_b, [K_s || ID_A || T])$   
 $B \rightarrow A: E(K_s, N_1)$   
 $A \rightarrow B: E(K_s, f(N_1))$

$T$  is a timestamp that assures  $A$  and  $B$  that the session key has only just been generated. Thus, both  $A$  and  $B$  know that the key distribution is a fresh exchange.

$A$  and  $B$  can verify time by checking that  $|Clock - T| < \Delta t_1 + \Delta t_2$ , where  $\Delta t_1$ , is the estimated normal difference between the KDC's clock and the local clock (at  $A$  or  $B$ ) and  $\Delta t_2$  is the expected network delay time.



## **Disadvantages of Denning solution**

A new concern is raised: namely, that this new scheme requires rely on clocks that are synchronized throughout the network points out a risk involved.

The risk is based on the fact that the distributed clocks can become unsynchronized as a result of damage on or faults in the clocks or the synchronization mechanism.

The problem occurs when a sender's clock is ahead of the intended recipient's clock.

In this case, an opponent can intercept a message from the sender and replay it later when the timestamp in the message becomes current at the recipient's site. This replay could cause unexpected results.

Gong refers to such attacks as suppress-replay attacks. One way to counter suppress-replay attacks is to enforce the requirement that parties regularly check their clocks against the KDC's clock.



## Alternate Solution for Suppress-replay attacks

The other alternative, which avoids the need for clock synchronization, is to rely on handshaking protocols using nonces. This alternative is not vulnerable to a suppress-replay attack, because the nonces the recipient will choose in the future are unpredictable to the sender.

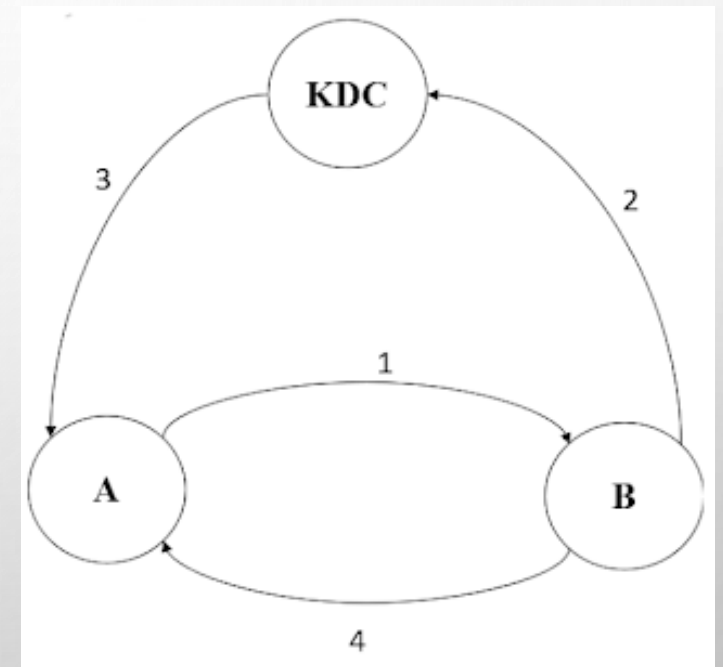
The Needham/Schroeder protocol relies on nonces only but, as we have seen, has other vulnerabilities. Improved strategy was presented in this protocol is:

$A \rightarrow B: ID_A || N_a$

$B \rightarrow KDC: ID_B || N_b || E(K_b, [ID_A || N_a || T_b])$

$KDC \rightarrow A: E(K_a, [ID_B || N_a || K_s || T_b]) || E(K_b, [ID_A || K_s || T_b] || N_b)$

$A \rightarrow B: E(K_b, [ID_A || K_s || T_b]) || E(K_s, N_b)$



Regenerating the session

**The following protocol ensures:**

$A \rightarrow B: E(K_b, [ID_A || K_s || T_b]) || N_a'$

$B \rightarrow A: N_b' || E(K_s, N_a')$

$A \rightarrow B: E(K_s, N_b')$

When B receives the message in step 1, it verifies that the ticket has not expired. The newly generated nonces and assure each party that there is no replay attack.



## One-way Authentication

With some refinement, the KDC strategy is a candidate for encrypted electronic mail. Because we wish to avoid requiring that the recipient (B) be on line at the same time as the sender (A), steps 4 and 5 must be eliminated. For a message with content, the sequence is as follows:

$$\begin{aligned} A &\rightarrow KDC: ID_A || ID_B || N_1 \\ KDC &\rightarrow A: E(K_a, [K_s || ID_B || N_1 || E(K_b, [K_s || ID_A])]) \\ A &\rightarrow B: E(K_b, [K_s || ID_A]) || E(K_s, M) \end{aligned}$$

This approach guarantees that only the intended recipient of a message will be able to read it. It also provides a level of authentication that the sender is A. The protocol does not protect against replay attack.

