

Needham Schroeder Symmetric Key

Author(s): Roger Needham and Michael Schroeder 1978

Last modified November 8, 2002

Summary: Distribution of a shared symmetric key by a trusted server and mutual authentication. Symmetric key cryptography with server.

Protocol specification (in common syntax)

```
A, B, S :      principal
Na, Nb :      nonce
Kas, Kbs, Kab : key
dec :         nonce -> nonce

1.   A  -> S   :   A, B, Na
2.   S  -> A   :   {Na, B, Kab, {Kab, A}Kbs}Kas
3.   A  -> B   :   {Kab, A}Kbs
4.   B  -> A   :   {Nb}Kab
5.   A  -> B   :   {dec(Nb)}Kab
```

Description of the protocol rules

This protocol establishes the fresh shared symmetric key K_{ab} .

Messages 1-3 perform the distribution of the fresh shared symmetric key K_{ab} and messages 4-5 are for mutual authentication of A and B.

The operator `dec` is decrementation.

Requirements

The protocol must guaranty the secrecy of K_{ab} : in every session, the value of K_{ab} must be known only by the participants playing the roles of A, B and S in that session.

If the participant playing B accepts the last message 5, then K_{ab} has been sent in message 3. by A (whose identity is included in the cipher of message 3).

References

[NS78].

Claimed attacks

Authentication attack by Denning and Sacco [DS81]. Assume that I has recorded the session i and that K_{ab} is compromised. After the session ii, B is convinced that he shares the secret key K_{ab} only with A.

i.1.	A	->	S	:	A, B, N_a
i.2.	S	->	A	:	$\{N_a, B, K_{ab}, \{K_{ab}, A\}_{K_{bs}}\}_{K_{as}}$
i.3.	A	->	I(B)	:	$\{K_{ab}, A\}_{K_{bs}}$
					assume that K_{ab} is compromised
ii.3.	I(A)	->	B	:	$\{K_{ab}, A\}_{K_{bs}}$
ii.4.	B	->	I(A)	:	$\{N_b\}_{K_{ab}}$
ii.5.	I(A)	->	B	:	$\{\text{dec}(N_b)\}_{K_{ab}}$

See also

Amended Needham Schroeder Symmetric Key, Denning-Sacco shared key, Kerberos V5.

Citations

- [DS81] D. Denning and G. Sacco. Timestamps in key distributed protocols. *Communication of the ACM*, 24(8):533–535, 1981.
- [NS78] R. Needham and M. Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12), December 1978.