Foundations of Computer Security

Lecture 65: The BAN Logic: Needham-Schroeder

Dr. Bill Young
Department of Computer Sciences
University of Texas at Austin

Needham-Schroeder: Idealization

Recall the Needham-Schroeder protocol:

1. $A \rightarrow S : A, B, N_a$
2. $S \rightarrow A : \{N_a, B, K_{ab}, \{K_{ab}, A\}_{K_{bs}}\}_{K_{as}}$
3. $A \rightarrow B : \{K_{ab}, A\}_{K_{bs}}$
4. $B \rightarrow A : \{N_b\}_{K_{ab}}$
5. $A \rightarrow B : \{N_b - 1\}_{K_{ab}}$

Needham-Schroeder is idealized as follows:

1. omitted since all components are plaintext
2. $S \rightarrow A : \{N_a, (A \overset{K_{ab}}{\longleftrightarrow} B), \#(A \overset{K_{ab}}{\longleftrightarrow} B), \{A \overset{K_{ab}}{\longleftrightarrow} B\}_{K_{bs}}\}_{K_{as}}$
3. $A \rightarrow B : \{A \overset{K_{ab}}{\longleftrightarrow} B\}_{K_{bs}}$
4. $B \rightarrow A : \{N_b, (A \overset{K_{ab}}{\longleftrightarrow} B)\}_{K_{ab}}$ *from B*
5. $A \rightarrow B : \{N_b, (A \overset{K_{ab}}{\longleftrightarrow} B)\}_{K_{ab}}$ *from A*

BAN Logic: Assumptions

The following initial assumptions are given for Needham-Schroeder:

$A| \equiv A \xleftrightarrow{K_{as}} S \qquad B| \equiv B \xleftrightarrow{K_{bs}} S \qquad S| \equiv A \xleftrightarrow{K_{as}} S$

$S| \equiv B \xleftrightarrow{K_{bs}} S$

$S| \equiv A \xleftrightarrow{K_{ab}} B$

$A| \equiv (S \Longrightarrow A \xleftrightarrow{K} B) \qquad B| \equiv (S \Longrightarrow A \xleftrightarrow{K} B)$

$A| \equiv (S \Longrightarrow \#(A \xleftrightarrow{K} B))$

$A| \equiv \#(N_a) \qquad B| \equiv \#(N_b) \qquad S| \equiv \#(A \xleftrightarrow{K_{ab}} B)$

$B| \equiv \#(A \xleftrightarrow{K} B)$

The very last of these is pretty strong. Needham and Schroeder did not realize they were making it, and were criticized for it.

BAN Logic: Analyzing the Protocol

From step 2 of the (idealized) protocol:

$$A \triangleleft \{N_a, (A \overset{K_{ab}}{\longleftrightarrow} B), \#(A \overset{K_{ab}}{\longleftrightarrow} B), \{A \overset{K_{ab}}{\longleftrightarrow} B\}_{K_{bs}}\}_{K_{as}}$$

The *Nonce Verification Rule* says:

$$\frac{A| \equiv (\#(X)), A| \equiv (S|\sim X)}{A| \equiv (S| \equiv X)}$$

Since A believes N_a to be fresh, we get:

$$A| \equiv (S| \equiv A \overset{K_{ab}}{\longleftrightarrow} B)$$

The *Jurisdiction Rule* says that:

$$\frac{A|\equiv (S \Longrightarrow X), A|\equiv (S|\equiv X)}{A|\equiv X}$$

From this we obtain:

$$A|\equiv A \stackrel{K_{ab}}{\longleftrightarrow} B$$

$$A|\equiv \#(A \stackrel{K_{ab}}{\longleftrightarrow} B)$$

BAN Logic: Analyzing the Protocol

Since A has also seen the part of the message encrypted under B's key, he can send it to B. B decrypts the message and obtains:

$$B \mid\equiv (S \mid\sim A \xleftrightarrow{K_{ab}} B)$$

meaning that B believes that S once sent the key.

At this point, we need the final dubious assumption:

$$B \mid\equiv \#(A \xleftrightarrow{K} B)$$

With it, we can get:

$$B \mid\equiv A \xleftrightarrow{K_{ab}} B$$

BAN Logic: Analyzing the Protocol

From the last two messages, we can infer the following. How?

$$A \mid\equiv A \stackrel{K_{ab}}{\longleftrightarrow} B$$

$$B \mid\equiv A \stackrel{K_{ab}}{\longleftrightarrow} B$$

$$A \mid\equiv (B \mid\equiv A \stackrel{K_{ab}}{\longleftrightarrow} B)$$

$$B \mid\equiv (A \mid\equiv A \stackrel{K_{ab}}{\longleftrightarrow} B)$$

These are the point of the protocol. The proof exhibits some assumptions that were not apparent.

Lessons

- Use of a logic like BAN shows what is provable and also what must be assumed.
- Using BAN effectively requires a lot of practice and insight into the protocol.

Next lecture: PGP