Protocol Verification Techniques - Belief Logics

Design and Verification of Security Protocols and Security Ceremonies

Programa de Pós-Graduacão em Ciências da Computacão
Dr. Jean Everson Martina

August-November 2016

Programa de
Pós-Graduação
em Ciência da
Computação
UFSC

UNIVERSIDADE FEDERAL
DE SANTA CATARINA

Disclaimer

Disclaimer!

This lecture is heavily based on material Professor Ravi Sandhu's from University of Texas San Antonio, and from material from "Paul Syverson and Iliano Cervesato, The Logic of Authentication Protocols, Foundations of Security Analysis and Design, LNCS 2171, SpringerVerlag, 2001."

BAN Logic

- BAN is a logic of belief;

BAN Logic

- BAN is a logic of belief;
- In an analysis, the protocol is first idealised into messages containing assertions;

BAN Logic

- BAN is a logic of belief;
- In an analysis, the protocol is first idealised into messages containing assertions;
- Then assumptions are stated;

BAN Logic

- BAN is a logic of belief;
- In an analysis, the protocol is first idealised into messages containing assertions;
- Then assumptions are stated;
- Finally conclusions are inferred based on the assertions in the idealized messages and those assumptions.

The language of BAN

- In all of these expressions, X is either a message or a formula;

The language of BAN

- In all of these expressions, X is either a message or a formula;
- As we will see, every formula can be a message, but not every message is a formula.

The language of BAN

- P believes X :

The language of BAN

- P believes X :
- P received X :

The language of BAN

- P believes X :
- P received X :
- P said X :

The language of BAN

- P believes X :
- P received X :
- P said X :
- P controls X :

The language of BAN

- P believes X :
- P received X :
- P said X :
- P controls X :
- fresh(X) :

The language of BAN

- P believes X :
- P received X :
- P said X :
- P controls X :
- fresh(X) :

The language of BAN

- $P \overset{k}{\longleftrightarrow} Q$:

The language of BAN

- P $\overset{k}{\leftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);

The language of BAN

- P $\overset{k}{\leftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);
- PK(P, k) :

The language of BAN

- P $\overset{k}{\longleftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);
- PK(P, k) :
  - The secret key, k^{-1}, corresponding to k will never be discovered by any principal but P or a principal trusted by P;

The language of BAN

- P $\overset{k}{\leftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);
- PK(P, k) :
  - The secret key, k^{-1}, corresponding to k will never be discovered by any principal but P or a principal trusted by P;
- $\{X\}_k$:

The language of BAN

- P $\overset{k}{\leftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);
- PK(P, k) :
  - The secret key, k^{-1}, corresponding to k will never be discovered by any principal but P or a principal trusted by P;
- $\{X\}_k$:
  - This is the notation for encryption;

The language of BAN

- P $\overset{k}{\leftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);
- PK(P, k) :
  - The secret key, k^{-1}, corresponding to k will never be discovered by any principal but P or a principal trusted by P;
- $\{X\}_k$:
  - This is the notation for encryption;
  - Principals can recognize their own messages;

The language of BAN

- P $\overset{k}{\leftrightarrow}$ Q :
  - k will never be discovered by any principal but P, Q, or a principal trusted by P or Q. (The last case is necessary, since the server often sees, indeed generates, k.);
- PK(P, k) :
  - The secret key, k^{-1}, corresponding to k will never be discovered by any principal but P or a principal trusted by P;
- $\{X\}_k$:
  - This is the notation for encryption;
  - Principals can recognize their own messages;
  - Encrypted messages are uniquely readable and verifiable as such by holders of the right keys.

BAN Rules: Message Meaning

$$\frac{\begin{array}{c} P \quad \textit{believes} \quad P \overset{k}{\leftrightarrow} Q \\ P \quad \textit{received} \quad \{X\}_k \end{array}}{P \quad \textit{believes} \quad Q \quad \textit{said} \quad X}$$

BAN Rules: Message Meaning

$$\frac{\begin{array}{c} P \quad believes \quad P \overset{k}{\leftrightarrow} Q \\ P \quad received \quad \{X\}_k \end{array}}{P \quad believes \quad Q \quad said \quad X}$$

- “If P receives X encrypted with k and if P believes k is a good key for talking with Q, then P believes Q once said X.”

BAN Rules: Message Meaning

$$\frac{\begin{array}{c} P \quad \mathit{believes} \quad PK(Q, k) \\ P \quad \mathit{received} \quad \{X\}_{k^{-1}} \end{array}}{P \quad \mathit{believes} \quad Q \quad \mathit{said} \quad X}$$

BAN Rules: Message Meaning

$$\frac{\begin{array}{c} P \quad \mathit{believes} \quad PK(Q,k) \\ P \quad \mathit{received} \quad \{X\}_{k^{-1}} \end{array}}{P \quad \mathit{believes} \quad Q \quad \mathit{said} \quad X}$$

- There is no explicit distinction between signing and encryption;

BAN Rules: Message Meaning

$$\frac{\begin{array}{c} P \quad \textit{believes} \quad PK(Q,k) \\ P \quad \textit{received} \quad \{X\}_{k^{-1}} \end{array}}{P \quad \textit{believes} \quad Q \quad \textit{said} \quad X}$$

- There is no explicit distinction between signing and encryption;
- Both are represented by $\{X\}_k$ or $\{X\}_{k^{-1}}$;

BAN Rules: Message Meaning

$$\frac{\begin{array}{c} P \quad believes \quad PK(Q,k) \\ P \quad received \quad \{X\}_{k^{-1}} \end{array}}{P \quad believes \quad Q \quad said \quad X}$$

- There is no explicit distinction between signing and encryption;
- Both are represented by $\{X\}_k$ or $\{X\}_{k^{-1}}$;
- The distinction is implicit in the notation for the key used: k or k-1.

BAN Rules: Nonce Verification

$$\frac{\begin{array}{c} P \quad \mathit{believes} \quad \mathit{fresh}(X) \\ P \quad \mathit{believes} \quad Q \quad \mathit{said} \quad X \end{array}}{P \quad \mathit{believes} \quad Q \quad \mathit{believes} \quad X}$$

BAN Rules: Nonce Verification

$$\frac{P \quad \mathit{believes} \quad \mathit{fresh}(X) \qquad P \quad \mathit{believes} \quad Q \quad \mathit{said} \quad X}{P \quad \mathit{believes} \quad Q \quad \mathit{believes} \quad X}$$

- This rule allows promotion from the past to the present (something said some time in the past to a present belief);

BAN Rules: Nonce Verification

$$\frac{\begin{array}{c} P \quad believes \quad fresh(X) \\ P \quad believes \quad Q \quad said \quad X \end{array}}{P \quad believes \quad Q \quad believes \quad X}$$

- This rule allows promotion from the past to the present (something said some time in the past to a present belief);
- In order to be applied, X should not contain any encrypted text.

BAN Rules: Jurisdiction

$$\frac{P \quad \textit{believes} \quad Q \quad \textit{controls} \quad X \qquad P \quad \textit{believes} \quad Q \quad \textit{believes} \quad X}{P \quad \textit{believes} \quad X}$$

BAN Rules: Jurisdiction

$$\frac{\begin{array}{c} P \quad believes \quad Q \quad controls \quad X \\ P \quad believes \quad Q \quad believes \quad X \end{array}}{P \quad believes \quad X}$$

- The jurisdiction rule allows inferences that a principal believes a key is good, even though it is a random string that he has never seen before.

BAN Rules: Belief Conjuncatenation

$$\frac{\begin{array}{ccc} P & \textit{believes} & X \\ P & \textit{believes} & Y \end{array}}{P \quad \textit{believes}(X, Y)}$$

BAN Rules: Belief Conjuncatenation

$$\frac{P \quad believes \quad X \qquad P \quad believes \quad Y}{P \quad believes(X, Y)}$$

- The obvious rules apply to beliefs concerning concatenations of messages/conjunctions of formulae;

BAN Rules: Belief Conjuncatenation

$$\frac{\begin{array}{c} P \quad \textit{believes} \quad X \\ P \quad \textit{believes} \quad Y \end{array}}{P \quad \textit{believes}(X, Y)}$$

- The obvious rules apply to beliefs concerning concatenations of messages/conjunctions of formulae;
- Concatenations of messages and conjunctions of formulae are both represented as (X,Y) in the above rules.

BAN Rules: Belief Conjuncatenation

$$\frac{P \quad \mathit{believes} \quad Q \quad \mathit{said}(X, Y)}{P \quad \mathit{believes} \quad Q \quad \mathit{said} \quad X}$$

BAN Rules: Belief Conjuncatenation

$$\frac{P \quad \mathit{believes} \quad Q \quad \mathit{said}(X, Y)}{P \quad \mathit{believes} \quad Q \quad \mathit{said} \quad X}$$

$$\frac{P \quad \mathit{believes} \quad Q \quad \mathit{believes} \quad (X, Y)}{P \quad \mathit{believes} \quad Q \quad \mathit{believes} \quad X}$$

BAN Rules: Freshness Conjuncatenation

$$\frac{P \quad \textit{believes} \quad \textit{fresh}(X)}{P \quad \textit{believes} \quad \textit{fresh}(X, Y)}$$

BAN Rules: Freshness Conjuncatenation

$$\frac{P \quad \textit{believes} \quad \textit{fresh}(X)}{P \quad \textit{believes} \quad \textit{fresh}(X, Y)}$$

- This is how nonces lend freshness to other messages in BAN.

BAN Rules: Receiving Rules

$$\frac{P \quad \textit{believes} \quad P \overset{k}{\leftrightarrow} Q \qquad P \quad \textit{received} \quad \{X\}_k}{P \quad \textit{received} \quad X}$$

BAN Rules: Receiving Rules

$$\frac{P \quad \textit{believes} \quad P \overset{k}{\longleftrightarrow} Q \qquad P \quad \textit{received} \quad \{X\}_k}{P \quad \textit{received} \quad X}$$

$$\frac{P \quad \textit{received} \quad (X, Y)}{P \quad \textit{received} \quad X}$$

BAN Rules: Receiving Rules

$$\frac{P \quad \textit{believes} \quad P \overset{k}{\longleftrightarrow} Q \qquad P \quad \textit{received} \quad \{X\}_k}{P \quad \textit{received} \quad X}$$

$$\frac{P \quad \textit{received} \quad (X, Y)}{P \quad \textit{received} \quad X}$$

- A principal receiving a message also receives sub-messages he can uncover.

BAN Protocol Analysis

1. Choose a protocol;

BAN Protocol Analysis

1. Choose a protocol;
2. Write assumptions about the initial state;

BAN Protocol Analysis

1. Choose a protocol;
2. Write assumptions about the initial state;
3. Annotate the protocol:

BAN Protocol Analysis

1. Choose a protocol;
2. Write assumptions about the initial state;
3. Annotate the protocol:
   - For each message transmission $P \rightarrow Q : M$ in the protocol, assert Q received M;

BAN Protocol Analysis

1. Choose a protocol;
2. Write assumptions about the initial state;
3. Annotate the protocol:
    - For each message transmission $P \rightarrow Q : M$ in the protocol, assert Q received M;
4. Use the logic to derive the beliefs held by protocol principals.

Chosen Protocol: Needham-Schroeder Shared Key Protocols

1. A → S: A, B, N_A

Chosen Protocol: Needham-Schroeder Shared Key Protocols

1. A $\rightarrow$ S: A, B, N_A
2. S $\rightarrow$ A: $\{N_A, B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$

Chosen Protocol: Needham-Schroeder Shared Key Protocols

1. A → S: A, B, N_A
2. S → A: $\{N_A, B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$
3. A → B: $\{K_{AB}, A\}_{K_{BS}}$

Chosen Protocol: Needham-Schroeder Shared Key Protocols

1. A → S: A, B, N_A
2. S → A: $\{N_A, B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$
3. A → B: $\{K_{AB}, A\}_{K_{BS}}$
4. B → A: $\{N_B\}_{K_{AB}}$

Chosen Protocol: Needham-Schroeder Shared Key Protocols

1. A → S: A, B, N_A
2. S → A: $\{N_A, B, K_{AB}, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$
3. A → B: $\{K_{AB}, A\}_{K_{BS}}$
4. B → A: $\{N_B\}_{K_{AB}}$
5. A → B: $\{N_B - 1\}_{K_{AB}}$

Idealized Needham-Schroeder Shared Key Protocol

2. S → A: $\{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}), \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S$

Idealized Needham-Schroeder Shared Key Protocol

2. S → A: $\{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}), \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S$

3. A → B: $\{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S$

Idealized Needham-Schroeder Shared Key Protocol

2. S → A: $\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, \mathit{fresh}(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}, \mathit{from} \quad S$
3. A → B: $\{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}, \mathit{from} \quad S$
4. B → A: $\{N_B, A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{AB}}, \mathit{from} \quad B$

Idealized Needham-Schroeder Shared Key Protocol

2. S → A: $\{N_A, A \xleftrightarrow{K_{AB}} B, fresh(K_{AB}), \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, from \quad S$
3. A → B: $\{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, from \quad S$
4. B → A: $\{N_B, A \xleftrightarrow{K_{AB}} B\}_{K_{AB}}, from \quad B$
5. A → B: $\{N_A, A \xleftrightarrow{K_{AB}} B\}_{K_{AB}}, from \quad A$

Idealized Needham-Schroeder Shared Key Protocol

- Plaintext is omitted;

Idealized Needham-Schroeder Shared Key Protocol

- Plaintext is omitted;
- It is assumed that principals recognize their own messages;

Idealized Needham-Schroeder Shared Key Protocol

- Plaintext is omitted;
- It is assumed that principals recognize their own messages;
- With a shared key, if a recipient can decrypt a message, she can tell who it is from;

Idealized Needham-Schroeder Shared Key Protocol

- Plaintext is omitted;
- It is assumed that principals recognize their own messages;
- With a shared key, if a recipient can decrypt a message, she can tell who it is from;
- What is inside the encrypted messages is also altered;

Idealized Needham-Schroeder Shared Key Protocol

- Plaintext is omitted;
- It is assumed that principals recognize their own messages;
- With a shared key, if a recipient can decrypt a message, she can tell who it is from;
- What is inside the encrypted messages is also altered;
- Specifically, the key k_{AB} is replaced by assertions about it;

Idealized Needham-Schroeder Shared Key Protocol

- Plaintext is omitted;
- It is assumed that principals recognize their own messages;
- With a shared key, if a recipient can decrypt a message, she can tell who it is from;
- What is inside the encrypted messages is also altered;
- Specifically, the key k_{AB} is replaced by assertions about it;
- Also in the last message $N_B - 1$ is changed to just N_B.

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad \textit{believes} \quad A \overset{K_{AS}}{\longleftrightarrow} S$

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad \textit{believes} \quad A \xleftrightarrow{K_{AS}} S$

P2 $B \quad \textit{believes} \quad B \xleftrightarrow{K_{BS}} S$

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad \textit{believes} \quad A \xleftrightarrow{K_{AS}} S$

P2 $B \quad \textit{believes} \quad B \xleftrightarrow{K_{BS}} S$

P3 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad believes \quad A \xleftrightarrow{K_{AS}} S$

P2 $B \quad believes \quad B \xleftrightarrow{K_{BS}} S$

P3 $A \quad believes \quad S \quad controls \quad A \xleftrightarrow{K_{AB}} B$

P4 $B \quad believes \quad S \quad controls \quad A \xleftrightarrow{K_{AB}} B$

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad \textit{believes} \quad A \xleftrightarrow{K_{AS}} S$

P2 $B \quad \textit{believes} \quad B \xleftrightarrow{K_{BS}} S$

P3 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

P4 $B \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

P5 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad \textit{fresh}(A \xleftrightarrow{K_{AB}} B)$

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad \textit{believes} \quad A \xleftrightarrow{K_{AS}} S$

P2 $B \quad \textit{believes} \quad B \xleftrightarrow{K_{BS}} S$

P3 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

P4 $B \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

P5 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad \textit{fresh}(A \xleftrightarrow{K_{AB}} B)$

P6 $A \quad \textit{believes} \quad \textit{fresh}(N_A)$

Needham-Schroeder Shared Key Protocol Assumptions

P1 $A \quad \textit{believes} \quad A \xleftrightarrow{K_{AS}} S$

P2 $B \quad \textit{believes} \quad B \xleftrightarrow{K_{BS}} S$

P3 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

P4 $B \quad \textit{believes} \quad S \quad \textit{controls} \quad A \xleftrightarrow{K_{AB}} B$

P5 $A \quad \textit{believes} \quad S \quad \textit{controls} \quad \textit{fresh}(A \xleftrightarrow{K_{AB}} B)$

P6 $A \quad \textit{believes} \quad \textit{fresh}(N_A)$

P7 $B \quad \textit{believes} \quad \textit{fresh}(N_B)$

Needham-Schroeder Shared Key Protocol Assumptions

- P1, P2 are beliefs in quality of long term keys;

Needham-Schroeder Shared Key Protocol Assumptions

- P1, P2 are beliefs in quality of long term keys;
- S has similar beliefs but are not relevant;

Needham-Schroeder Shared Key Protocol Assumptions

- P1, P2 are beliefs in quality of long term keys;
- S has similar beliefs but are not relevant;
- P3, P4, P5 are jurisdiction beliefs;

Needham-Schroeder Shared Key Protocol Assumptions

- P1, P2 are beliefs in quality of long term keys;
- S has similar beliefs but are not relevant;
- P3, P4, P5 are jurisdiction beliefs;
- P6, P7 are beliefs in freshness of each principal's nonces.

Needham-Schroeder Shared Key Protocol Annotations

P8 $A \quad \mathit{recieved} \quad \{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}),$
$\{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S$

Needham-Schroeder Shared Key Protocol Annotations

$$\begin{array}{ll} \text{P8} & A \quad \mathit{recieved} \quad \{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}), \\ & \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S \\ \text{P9} & B \quad \mathit{recieved} \quad \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S \end{array}$$

Needham-Schroeder Shared Key Protocol Annotations

P8 $A \quad \textit{recieved} \quad \{N_A, A \xleftrightarrow{K_{AB}} B, \textit{fresh}(K_{AB}),$
$\{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \textit{from} \quad S$

P9 $B \quad \textit{recieved} \quad \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \textit{from} \quad S$

P10 $A \quad \textit{recieved} \quad \{N_B, A \xleftrightarrow{K_{AB}} B\}_{K_{AB}}, \textit{from} \quad B$

Needham-Schroeder Shared Key Protocol Annotations

P8 $A \quad \mathit{recieved} \quad \{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}),$
$\{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S$

P9 $B \quad \mathit{recieved} \quad \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}, \mathit{from} \quad S$

P10 $A \quad \mathit{recieved} \quad \{N_B, A \xleftrightarrow{K_{AB}} B\}_{K_{AB}}, \mathit{from} \quad B$

P11 $B \quad \mathit{recieved} \quad \{N_A, A \xleftrightarrow{K_{AB}} B\}_{K_{AB}}, \mathit{from} \quad A$

Needham-Schroeder Shared Key Protocol Derivations

1. A believes S said $((\{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}), \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}));$

Needham-Schroeder Shared Key Protocol Derivations

1. A believes S said $((\{N_A, A \xleftrightarrow{K_{AB}} B, \mathit{fresh}(K_{AB}), \{A \xleftrightarrow{K_{AB}} B\}_{K_{BS}}))$;
   - By Message Meaning using P1, P8;

Needham-Schroeder Shared Key Protocol Derivations

1. A believes S said $((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;
   - By Message Meaning using P1, P8;
2. A believes fresh$((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;

Needham-Schroeder Shared Key Protocol Derivations

1. A believes S said $(((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;
    - By Message Meaning using P1, P8;
2. A believes fresh$((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;
    - By Freshness Conjuncatenation using 1, P6;

Needham-Schroeder Shared Key Protocol Derivations

1. A believes S said $((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, \mathit{fresh}(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;
   - By Message Meaning using P1, P8;
2. A believes fresh$((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, \mathit{fresh}(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;
   - By Freshness Conjuncatenation using 1, P6;
3. 3. A believes S believes $((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, \mathit{fresh}(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;

Needham-Schroeder Shared Key Protocol Derivations

1. A believes S said $((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}));$
   - By Message Meaning using P1, P8;
2. A believes fresh$((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}));$
   - By Freshness Conjuncatenation using 1, P6;
3. 3. A believes S believes $((\{N_A, A \overset{K_{AB}}{\longleftrightarrow} B, fresh(K_{AB}), \{A \overset{K_{AB}}{\longleftrightarrow} B\}_{K_{BS}}))$;
   - By Nonce Verification using 2, 1;

Needham-Schroeder Shared Key Protocol Derivations

4 A believes S believes $(A \xleftrightarrow{K_{AB}} B)$;

Needham-Schroeder Shared Key Protocol Derivations

4. A believes S believes ($A \xleftrightarrow{K_{AB}} B$);
   - By Belief Conjuncatenation using 3;

Needham-Schroeder Shared Key Protocol Derivations

4. A believes S believes $(A \xleftrightarrow{K_{AB}} B)$;
   - By Belief Conjuncatenation using 3;
5. A believes S believes fresh$(A \xleftrightarrow{K_{AB}} B)$;

Needham-Schroeder Shared Key Protocol Derivations

4. A believes S believes $(A \xleftrightarrow{K_{AB}} B)$;
   - By Belief Conjuncatenation using 3;
5. A believes S believes fresh$(A \xleftrightarrow{K_{AB}} B)$;
   - By Belief Conjuncatenation using 3;

Needham-Schroeder Shared Key Protocol Derivations

4. A believes S believes $(A \xleftrightarrow{K_{AB}} B)$;
    - By Belief Conjuncatenation using 3;
5. A believes S believes fresh$(A \xleftrightarrow{K_{AB}} B)$;
    - By Belief Conjuncatenation using 3;
6. A believes $(A \xleftrightarrow{K_{AB}} B)$;

Needham-Schroeder Shared Key Protocol Derivations

4. A believes S believes $(A \xleftrightarrow{K_{AB}} B)$;
   - By Belief Conjuncatenation using 3;
5. A believes S believes fresh$(A \xleftrightarrow{K_{AB}} B)$;
   - By Belief Conjuncatenation using 3;
6. A believes $(A \xleftrightarrow{K_{AB}} B)$;
   - By Jurisdiction using 4, P3;

Needham-Schroeder Shared Key Protocol Derivations

7. A believes fresh($A \xleftrightarrow{K_{AB}} B$);

Needham-Schroeder Shared Key Protocol Derivations

7. A believes fresh$(A \overset{K_{AB}}{\longleftrightarrow} B)$;
   - By Jurisdiction using 4, P5;

Needham-Schroeder Shared Key Protocol Derivations

7. A believes fresh($A \xleftrightarrow{K_{AB}} B$);
   - By Jurisdiction using 4, P5;

   Here we finished proving Alice's Beliefs. She believes K_{AB} is secure and fresh.

Needham-Schroeder Shared Key Protocol Derivations

7 A believes fresh($A \xleftrightarrow{K_{AB}} B$);

- By Jurisdiction using 4, P5;

Here we finished proving Alice's Beliefs. She believes K_{AB} is secure and fresh.

8 B believes S said ($A \xleftrightarrow{K_{AB}} B$);

Needham-Schroeder Shared Key Protocol Derivations

7. A believes fresh($A \xleftrightarrow{K_{AB}} B$);
   - By Jurisdiction using 4, P5;

   Here we finished proving Alice's Beliefs. She believes K_{AB} is secure and fresh.

8. B believes S said ($A \xleftrightarrow{K_{AB}} B$);
   - By Message Meaning using P2, P9;

Needham-Schroeder Shared Key Protocol Derivations

7. A believes fresh($A \xleftrightarrow{K_{AB}} B$);
    - By Jurisdiction using 4, P5;

   Here we finished proving Alice's Beliefs. She believes K_{AB} is secure and fresh.

8. B believes S said ($A \xleftrightarrow{K_{AB}} B$);
    - By Message Meaning using P2, P9;

   Bob believes K_{AB} is secure, but has nothing regarding freshness.

Needham-Schroeder Shared Key Protocol Derivations

We need to introduce a new Assumption:

Needham-Schroeder Shared Key Protocol Derivations

We need to introduce a new Assumption:

$$\text{P12} \quad B \quad \textit{believes} \quad \textit{fresh}(A \xleftrightarrow{K_{AB}} B)$$

Needham-Schroeder Shared Key Protocol Derivations

We need to introduce a new Assumption:

$$\text{P12} \quad B \quad \textit{believes} \quad \textit{fresh}(A \xleftrightarrow{K_{AB}} B)$$

Attention!

This is sketchy, since Bob believes that a random value generated by someone else is fresh.

Needham-Schroeder Shared Key Protocol Derivations

9 B believes S believes ($A \xleftrightarrow{K_{AB}} B$);

Needham-Schroeder Shared Key Protocol Derivations

9. B believes S believes $(A \xleftrightarrow{K_{AB}} B)$;
   - By Nonce Verification using P12, 8;

Needham-Schroeder Shared Key Protocol Derivations

9 B believes S believes ($A \xleftrightarrow{K_{AB}} B$);

- By Nonce Verification using P12, 8;

10 B believes $A \xleftrightarrow{K_{AB}} B$;

Needham-Schroeder Shared Key Protocol Derivations

9 B believes S believes $(A \xleftrightarrow{K_{AB}} B)$;
- By Nonce Verification using P12, 8;

10 B believes $A \xleftrightarrow{K_{AB}} B$;
- By Jurisdiction using P4, 9.

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Message Meaning using 6, P10;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Message Meaning using 6, P10;

12 A believes fresh$(N_B, A \xleftrightarrow{K_{AB}} B)$;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Message Meaning using 6, P10;

12 A believes fresh$(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Freshness Conjuncatenation using 7;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Message Meaning using 6, P10;

12 A believes fresh$(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Freshness Conjuncatenation using 7;

13 A believes B believes $(N_B, A \xleftrightarrow{K_{AB}} B)$;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Message Meaning using 6, P10;

12 A believes fresh$(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Freshness Conjuncatenation using 7;

13 A believes B believes $(N_B, A \xleftrightarrow{K_{AB}} B)$;

- By Nonce Verification using 12, 11;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \overset{K_{AB}}{\longleftrightarrow} B)$;
   - By Message Meaning using 6, P10;

12 A believes fresh$(N_B, A \overset{K_{AB}}{\longleftrightarrow} B)$;
   - By Freshness Conjuncatenation using 7;

13 A believes B believes $(N_B, A \overset{K_{AB}}{\longleftrightarrow} B)$;
   - By Nonce Verification using 12, 11;

14 A believes B believes $(A \overset{K_{AB}}{\longleftrightarrow} B)$;

Needham-Schroeder Shared Key Protocol Derivations

11 A believes B said $(N_B, A \xleftrightarrow{K_{AB}} B)$;
- By Message Meaning using 6, P10;

12 A believes fresh$(N_B, A \xleftrightarrow{K_{AB}} B)$;
- By Freshness Conjuncatenation using 7;

13 A believes B believes $(N_B, A \xleftrightarrow{K_{AB}} B)$;
- By Nonce Verification using 12, 11;

14 A believes B believes $(A \xleftrightarrow{K_{AB}} B)$;
- By Belief Conjuncatenation using 13.

BAN's Limitations

- BAN logic assumes that agents never publish secrets, but BAN logic does not verify this;

BAN's Limitations

- BAN logic assumes that agents never publish secrets, but BAN logic does not verify this;
- BAN logic assumes that agents can recognize type flaws, but BAN logic does not verify the absence of type flaws;

BAN's Limitations

- BAN logic assumes that agents never publish secrets, but BAN logic does not verify this;
- BAN logic assumes that agents can recognize type flaws, but BAN logic does not verify the absence of type flaws;
- In particular, BAN logic assumes that agents always recognize and ignore messages that they have sent themselves;

BAN's Limitations

- BAN logic assumes that agents never publish secrets, but BAN logic does not verify this;
- BAN logic assumes that agents can recognize type flaws, but BAN logic does not verify the absence of type flaws;
- In particular, BAN logic assumes that agents always recognize and ignore messages that they have sent themselves;
- BAN logic assumes that all protocol participants are honest. No compromised agents are considered. Attackers do not have valid keys.

Needham's Quotation

Roger Needham

"The main contribution of BAN logic was to make the study of 3-line protocols intellectually respectable."

Questions????

UNIVERSIDADE FEDERAL
DE SANTA CATARINA

UNIVERSIDADE FEDERAL
DE SANTA CATARINA