We further require rules for intruder actions.

$$\begin{aligned} I(x), I(y) &\leadsto I(\langle x, y\rangle), I(x), I(y) \\ I(\langle x, y\rangle) &\leadsto I(x), I(y), I(\langle x, y\rangle) \end{aligned}$$

All our rules never consume I facts, although the rules may create I facts.

I is said to be persistent.

This represents the intruder's ability to remember all previous messages, and use them again and again to compute new messages.

In contrast our previous rules consume A_0 facts to create A_1 facts, consume A_1 facts to create A_2 facts, . . .

To start execution, we also need the initialization rules.

$$\leadsto A_0() \qquad \text{and} \qquad \leadsto B_0$$

To start execution, we also need the initialization rules.

$$\leadsto A_0() \qquad \text{and} \qquad \leadsto B_0$$

An execution with involving more than one session:

$A_0()$

To start execution, we also need the initialization rules.

$\rightsquigarrow A_0()$ and $\rightsquigarrow B_0$

An execution with involving more than one session:

$A_0()$

$\rightsquigarrow B_0(), A_0()$

To start execution, we also need the initialization rules.

$\rightsquigarrow A_0()$ and $\rightsquigarrow B_0$

An execution with involving more than one session:

$A_0()$

$\rightsquigarrow B_0(), A_0()$

$\rightsquigarrow A_1(n_a), I(n_a), B_0()$

To start execution, we also need the initialization rules.

$$\rightsquigarrow A_0() \qquad \text{and} \qquad \rightsquigarrow B_0$$

An execution with involving more than one session:

$A_0()$

$\rightsquigarrow B_0(), A_0()$

$\rightsquigarrow A_1(n_a), I(n_a), B_0()$

$\rightsquigarrow B_1(n_a, n_b), I(\langle n_a, n_b \rangle), I(n_a), A_1(n_a)$

To start execution, we also need the initialization rules.

$\rightsquigarrow A_0()$ and $\rightsquigarrow B_0$

An execution with involving more than one session:

$A_0()$

$\rightsquigarrow B_0(), A_0()$

$\rightsquigarrow A_1(n_a), I(n_a), B_0()$

$\rightsquigarrow B_1(n_a, n_b), I(\langle n_a, n_b\rangle), I(n_a), A_1(n_a)$

$\ldots \rightsquigarrow A_1(n'_a), I(n'_a), B_0(), \quad B_1(n_a, n_b), I(\langle n_a, n_b\rangle), I(n_a), A_1(n_a)$

(A new session)

To start execution, we also need the initialization rules.

$\rightsquigarrow A_0()$ and $\rightsquigarrow B_0$

An execution with involving more than one session:

$A_0()$

$\rightsquigarrow B_0(), A_0()$

$\rightsquigarrow A_1(n_a), I(n_a), B_0()$

$\rightsquigarrow B_1(n_a, n_b), I(\langle n_a, n_b \rangle), I(n_a), A_1(n_a)$

$\ldots \rightsquigarrow A_1(n'_a), I(n'_a), B_0(), \quad B_1(n_a, n_b), I(\langle n_a, n_b \rangle), I(n_a), A_1(n_a)$

(A new session)

$\ldots \rightsquigarrow A_1(n'_a), B_0(), B_1(n_a, n_b), A_1(n_a), I(\langle n'_a, n_b \rangle), \ldots$

(Intruder actions)

To start execution, we also need the initialization rules.

$$\rightsquigarrow A_0() \qquad \text{and} \qquad \rightsquigarrow B_0$$

An execution with involving more than one session:

$A_0()$

$\rightsquigarrow B_0(), A_0()$

$\rightsquigarrow A_1(n_a), I(n_a), B_0()$

$\rightsquigarrow B_1(n_a, n_b), I(\langle n_a, n_b \rangle), I(n_a), A_1(n_a)$

$\ldots \rightsquigarrow A_1(n'_a), I(n'_a), B_0(), \quad B_1(n_a, n_b), I(\langle n_a, n_b \rangle), I(n_a), A_1(n_a)$

(A new session)

$\ldots \rightsquigarrow A_1(n'_a), B_0(), B_1(n_a, n_b), A_1(n_a), I(\langle n'_a, n_b \rangle), \ldots$

(Intruder actions)

$\rightsquigarrow A_2(n'_a, n_b), B_0(), B_1(n_a, n_b), A_1(n_a), \ldots$

The full protocol

$$A \longrightarrow B : \{a, n_a\}_{k_b}$$
$$B \longrightarrow A : \{n_a, n_b\}_{k_a}$$
$$A \longrightarrow B : \{n_b\}_{k_b}$$

List of honest agents

$\leadsto Ha(alice)$ $\leadsto Ha(bob)$ $\leadsto Ha(michael)$ $\ldots$

List of dishonest agents

$\leadsto Da(charlie)$ $\ldots$

$Ha(x) \leadsto Agent(x), Ha(x)$ $Da(x) \leadsto Agent(x), Da(x)$

Their names and keys

$Agent(x) \leadsto I(x), I(pub(x)), Agent(x)$ $Da(x) \leadsto I(prv(x)), Da(x)$

Intruder actions

$$\begin{aligned}
I(x), I(y) &\leadsto I(\langle x, y\rangle), I(x), I(y) \\
I(\langle x, y\rangle) &\leadsto I(x), I(y), I(\langle x, y\rangle) \\
I(x), I(y) &\leadsto I(\{x\}_y), I(x), I(y) \\
I(\{x\}_{pub(y)}), I(prv(y)) &\leadsto I(x), I(\{x\}_{pub(y)}), I(prv(y)) \\
I(\{x\}_{prv(y)}), I(pub(y)) &\leadsto I(x), I(\{x\}_{prv(y)}), I(pub(y)) \\
I(x), I(y) &\leadsto I(\{x\}_y), I(x), I(y) \\
I(\{x\}_y), I(y) &\leadsto I(x), I(\{x\}_y), I(y)
\end{aligned}$$

($\{\_\}\_$ denotes symmetric encryption)

$$\leadsto \exists n \cdot I(n)$$

(Intruder may create nonces)

Protocol dependent rules

$$\begin{aligned} Agent(x), Agent(y) &\quad \rightsquigarrow A_0(x, y), Agent(x), Agent(y) \\ Agent(x), Agent(y) &\quad \rightsquigarrow B_0(x, y), Agent(x), Agent(y) \end{aligned}$$

Protocol dependent rules

$$\begin{aligned}
Agent(x), Agent(y) &\leadsto A_0(x, y), Agent(x), Agent(y) \\
Agent(x), Agent(y) &\leadsto B_0(x, y), Agent(x), Agent(y) \\
A_0(x, y) &\leadsto \exists z \cdot A_1(x, y, z), I(\{x, z\}_{pub(y)})
\end{aligned}$$

Protocol dependent rules

$$\begin{aligned}
Agent(x), Agent(y) &\leadsto A_0(x, y), Agent(x), Agent(y) \\
Agent(x), Agent(y) &\leadsto B_0(x, y), Agent(x), Agent(y) \\
A_0(x, y) &\leadsto \exists z \cdot A_1(x, y, z), I(\{x, z\}_{pub(y)}) \\
B_0(x, y), I(\{x, z\}_{pub(y)}) &\leadsto \exists w \cdot B_1(x, y, z, w), I(\{z, w\}_{pub(x)}), \\
&\qquad I(\{x, z\}_{pub(y)})
\end{aligned}$$

Protocol dependent rules

$$
\begin{aligned}
Agent(x), Agent(y) &\rightsquigarrow A_0(x, y), Agent(x), Agent(y) \\
Agent(x), Agent(y) &\rightsquigarrow B_0(x, y), Agent(x), Agent(y) \\
A_0(x, y) &\rightsquigarrow \exists z \cdot A_1(x, y, z), I(\{x, z\}_{pub(y)}) \\
B_0(x, y), I(\{x, z\}_{pub(y)}) &\rightsquigarrow \exists w \cdot B_1(x, y, z, w), I(\{z, w\}_{pub(x)}), \\
&\qquad I(\{x, z\}_{pub(y)}) \\
A_1(x, y, z), I(\{z, w\}_{pub(x)}) &\rightsquigarrow A_2(x, y, z, w), I(\{w\}_{pub(y)}), \\
&\qquad I(\{z, w\}_{pub(x)})
\end{aligned}
$$

Protocol dependent rules

$$
\begin{aligned}
Agent(x), Agent(y) &\leadsto A_0(x, y), Agent(x), Agent(y) \\
Agent(x), Agent(y) &\leadsto B_0(x, y), Agent(x), Agent(y) \\
A_0(x, y) &\leadsto \exists z \cdot A_1(x, y, z), I(\{x, z\}_{pub(y)}) \\
B_0(x, y), I(\{x, z\}_{pub(y)}) &\leadsto \exists w \cdot B_1(x, y, z, w), I(\{z, w\}_{pub(x)}), \\
&\qquad I(\{x, z\}_{pub(y)}) \\
A_1(x, y, z), I(\{z, w\}_{pub(x)}) &\leadsto A_2(x, y, z, w), I(\{w\}_{pub(y)}), \\
&\qquad I(\{z, w\}_{pub(x)}) \\
B_1(x, y, z, w), I(\{w\}_{pub(y)}) &\leadsto B_2(x, y, z, w), I(\{w\}_{pub(y)})
\end{aligned}
$$

Protocol dependent rules

$$\begin{aligned}
Agent(x), Agent(y) &\leadsto A_0(x,y), Agent(x), Agent(y)\\
Agent(x), Agent(y) &\leadsto B_0(x,y), Agent(x), Agent(y)\\
A_0(x,y) &\leadsto \exists z \cdot A_1(x,y,z), I(\{x,z\}_{pub(y)})\\
B_0(x,y), I(\{x,z\}_{pub(y)}) &\leadsto \exists w \cdot B_1(x,y,z,w), I(\{z,w\}_{pub(x)}),\\
&\qquad I(\{x,z\}_{pub(y)})\\
A_1(x,y,z), I(\{z,w\}_{pub(x)}) &\leadsto A_2(x,y,z,w), I(\{w\}_{pub(y)}),\\
&\qquad I(\{z,w\}_{pub(x)})\\
B_1(x,y,z,w), I(\{w\}_{pub(y)}) &\leadsto B_2(x,y,z,w), I(\{w\}_{pub(y)})
\end{aligned}$$

$\Rightarrow$ Protocol rules are independent of names of agents

Example security questions

- From point of view of initiator:
    - Can one reach a protocol state which contains the pattern $$Ha(x), Ha(y), A_2(x, y, z, w), I(z)$$
    - Can one reach a protocol state which contains the pattern $$Ha(x), Ha(y), A_2(x, y, z, w), I(w)$$
- Reachability questions regarding certain patterns.
- Independent of names of agents.
- Some agents are restricted to be honest.

Reduction of number of agents

- Shown by V. Cortier and H. Comon-Lundh.
- Protocol description and security (reachability) properties should be independent of agent names.
- Then we use our idea of projection as for ping-pong protocols.
- Replace all honest agents by one honest agent and all dishonest agents by one dishonest agent. Hence only two agents suffice.
- If agents cannot speak to themselves then a slightly higher bound can be shown.

Reduction to two agents

Suppose there is an attack which involves honest agents $h_1, \ldots, h_p$ and dishonest agents $d_1, \ldots, d_q$.

Define

$$\begin{aligned}
\mathtt{proj}(h_i) &= h_1 \\
\mathtt{proj}(d_j) &= d_1 \\
\mathtt{proj}(\{m_1\}_{m_2}) &= \{\mathtt{proj}(m_1)\}_{\mathtt{proj}(m_2)} \\
\mathtt{proj}(pub(a)) &= pub(\mathtt{proj}(a)) \\
\mathtt{proj}(P(t_1, t_2, \ldots)) &= P(\mathtt{proj}(t_1), \mathtt{proj}(t_2), \ldots) \\
&\ldots
\end{aligned}$$

If we can apply one of our rules to obtain a new protocol state from an old protocol state as follows:

$$F_1, \ldots, F_m \rightsquigarrow F'_1, \ldots, F'_n$$

then it is also possible to apply a rule to change protocol state as follows:

$$\mathtt{proj}(F_1), \ldots, \mathtt{proj}(F_m) \rightsquigarrow \mathtt{proj}(F'_1), \ldots, \mathtt{proj}(F'_n)$$

If we can apply one of our rules to obtain a new protocol state from an old protocol state as follows:

$$F_1, \ldots, F_m \leadsto F'_1, \ldots, F'_n$$

then it is also possible to apply a rule to change protocol state as follows:

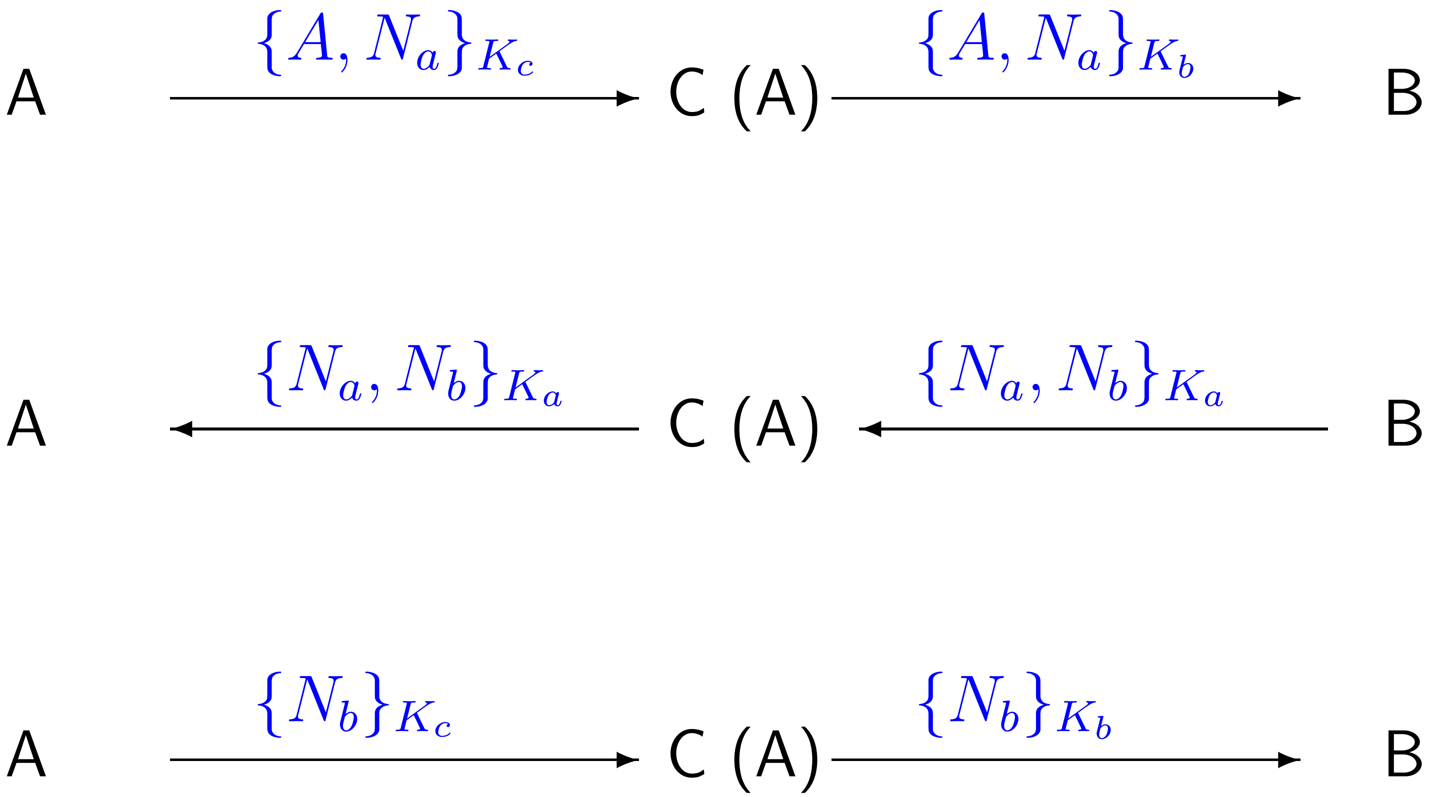
$$\texttt{proj}(F_1), \ldots, \texttt{proj}(F_m) \leadsto \texttt{proj}(F'_1), \ldots, \texttt{proj}(F'_n)$$

Hence if a protocol state $F_1, \ldots, F_n$ is reachable then the protocol state $\texttt{proj}(F_1), \ldots, \texttt{proj}(F_n)$ is also reachable.

⇒ Only two agents suffice for detecting an attack.

Example: attack on the Needham-Schroeder public key protocol

Attack using 3 agents:



A $\xrightarrow{\{A, N_a\}_{K_c}}$ C (A) $\xrightarrow{\{A, N_a\}_{K_b}}$ B

A $\xleftarrow{\{N_a, N_b\}_{K_a}}$ C (A) $\xleftarrow{\{N_a, N_b\}_{K_a}}$ B

A $\xrightarrow{\{N_b\}_{K_c}}$ C (A) $\xrightarrow{\{N_b\}_{K_b}}$ B

Example: attack on the Needham-Schroeder public key protocol

Reduction to 2 agents:

$$\begin{tikzcd}
\text{A} \arrow[r, "\{A, N_a\}_{K_c}"] & \text{C (A)} \arrow[r, "\{A, N_a\}_{K_a}"] & \text{A} \\
\text{A} & \text{C (A)} \arrow[l, "\{N_a, N'_a\}_{K_a}"'] & \text{A} \arrow[l, "\{N_a, N'_a\}_{K_a}"'] \\
\text{A} \arrow[r, "\{N'_a\}_{K_c}"] & \text{C (A)} \arrow[r, "\{N'_a\}_{K_a}"] & \text{A}
\end{tikzcd}$$

$$X \to Y\text{:} \quad \{M\}_{K_Y}, X$$

$$Y \to X\text{:} \quad \{M\}_{K_X}$$

We may formalize it using the following rules.

$$Agent(x), Agent(y) \quad \rightsquigarrow A_0(x, y), B_0(x, y), Agent(x), Agent(y)$$

$$X \to Y\text{:} \quad \{M\}_{K_Y}, X$$

$$Y \to X\text{:} \quad \{M\}_{K_X}$$

We may formalize it using the following rules.

$$\begin{aligned}
Agent(x), Agent(y) \quad &\rightsquigarrow A_0(x,y), B_0(x,y), Agent(x), Agent(y) \\
A_0(x,y) \quad &\rightsquigarrow \exists z \cdot A_1(x,y,z), I(\langle \{z\}_{pub(y)}, x\rangle)
\end{aligned}$$

$$X \to Y\colon \quad \{M\}_{K_Y}, X$$

$$Y \to X\colon \quad \{M\}_{K_X}$$

We may formalize it using the following rules.

$$\begin{aligned}
Agent(x), Agent(y) &\leadsto A_0(x,y), B_0(x,y), Agent(x), Agent(y)\\
A_0(x,y) &\leadsto \exists z \cdot A_1(x,y,z), I(\langle \{z\}_{pub(y)}, x\rangle)\\
B_0(x,y), I(\langle \{z\}_{pub(y)}, x\rangle) &\leadsto B_1(x,y,z), I(\{z\}_{pub(x)}), I(\langle \{z\}_{pub(y)}, x\rangle)
\end{aligned}$$

$$X \to Y\colon \quad \{M\}_{K_Y}, X$$

$$Y \to X\colon \quad \{M\}_{K_X}$$

We may formalize it using the following rules.

$$\begin{aligned}
Agent(x), Agent(y) &\leadsto A_0(x,y), B_0(x,y), Agent(x), Agent(y)\\
A_0(x,y) &\leadsto \exists z \cdot A_1(x,y,z), I(\langle\{z\}_{pub(y)}, x\rangle)\\
B_0(x,y), I(\langle\{z\}_{pub(y)}, x\rangle) &\leadsto B_1(x,y,z), I(\{z\}_{pub(x)}), I(\langle\{z\}_{pub(y)}, x\rangle)\\
A_1(x,y,z), I(\{z\}_{pub(x)}) &\leadsto A_2(x,y,z), I(\{z\}_{pub(x)})
\end{aligned}$$

$$X \to Y\colon \quad \{M\}_{K_Y}, X$$

$$Y \to X\colon \quad \{M\}_{K_X}$$

We may formalize it using the following rules.

$$\begin{aligned}
Agent(x), Agent(y) \quad &\leadsto A_0(x,y), B_0(x,y), Agent(x), Agent(y)\\
A_0(x,y) \quad &\leadsto \exists z \cdot A_1(x,y,z), I(\langle \{z\}_{pub(y)}, x\rangle)\\
B_0(x,y), I(\langle \{z\}_{pub(y)}, x\rangle) \quad &\leadsto B_1(x,y,z), I(\{z\}_{pub(x)}), I(\langle \{z\}_{pub(y)}, x\rangle)\\
A_1(x,y,z), I(\{z\}_{pub(x)}) \quad &\leadsto A_2(x,y,z), I(\{z\}_{pub(x)})
\end{aligned}$$

Security question: is a protocol state reachable containing the pattern

$$Ha(x), Ha(y), A_2(x,y,z), I(z)$$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a, b), B_0(a, b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a, b), B_0(a, b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_1(a, b, n), B_0(a, b), I(\langle\{n\}_{pub(b)}, a\rangle), Agent(a), Agent(b), Agent(c), \ldots$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a, b), B_0(a, b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_1(a, b, n), B_0(a, b), I(\langle\{n\}_{pub(b)}, a\rangle), Agent(a), Agent(b), Agent(c), \ldots$
- $A_1(a, b, n), B_0(a, b), I(\langle\{n\}_{pub(b)}, a\rangle), I(c), I(\langle\{n\}_{pub(b)}, c\rangle), \ldots$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a,b), B_0(a,b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, a\rangle), Agent(a), Agent(b), Agent(c), \ldots$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, a\rangle), I(c), I(\langle\{n\}_{pub(b)}, c\rangle), \ldots$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, c\rangle), B_0(c,b), A_0(c,b), \ldots$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a, b), B_0(a, b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_1(a, b, n), B_0(a, b), I(\langle \{n\}_{pub(b)}, a\rangle), Agent(a), Agent(b), Agent(c), \ldots$
- $A_1(a, b, n), B_0(a, b), I(\langle \{n\}_{pub(b)}, a\rangle), I(c), I(\langle \{n\}_{pub(b)}, c\rangle), \ldots$
- $A_1(a, b, n), B_0(a, b), I(\langle \{n\}_{pub(b)}, c\rangle), B_0(c, b), A_0(c, b), \ldots$
- $A_1(a, b, n), B_0(a, b), B_1(c, b, n), I(\{n\}_{pub(c)}), \ldots$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a,b), B_0(a,b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, a\rangle), Agent(a), Agent(b), Agent(c), \ldots$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, a\rangle), I(c), I(\langle\{n\}_{pub(b)}, c\rangle), \ldots$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, c\rangle), B_0(c,b), A_0(c,b), \ldots$
- $A_1(a,b,n), B_0(a,b), B_1(c,b,n), I(\{n\}_{pub(c)}), \ldots$
- $A_1(a,b,n), B_0(a,b), B_1(c,b,n), I(\{n\}_{pub(c)}), I(n), I(\{n\}_{pub(a)}), \ldots$

Attack involving 3 agents: some intermediate steps

- $Ha(a), Hb(b), Da(c)$
- $Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_0(a,b), B_0(a,b), Agent(a), Agent(b), Agent(c), Ha(a), Hb(b), Da(c)$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, a\rangle), Agent(a), Agent(b), Agent(c), \ldots$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, a\rangle), I(c), I(\langle\{n\}_{pub(b)}, c\rangle), \ldots$
- $A_1(a,b,n), B_0(a,b), I(\langle\{n\}_{pub(b)}, c\rangle), B_0(c,b), A_0(c,b), \ldots$
- $A_1(a,b,n), B_0(a,b), B_1(c,b,n), I(\{n\}_{pub(c)}), \ldots$
- $A_1(a,b,n), B_0(a,b), B_1(c,b,n), I(\{n\}_{pub(c)}), I(n), I(\{n\}_{pub(a)}), \ldots$
- $A_2(a,b,n), B_0(a,b), B_1(c,b,n), I(\{n\}_{pub(c)}), I(n), I(\{n\}_{pub(a)}), \ldots$

Reduction to 2 agents: some intermediate steps

- $Ha(a), Ha(a), Da(c)$
- $Agent(a), Agent(a), Agent(c), Ha(a), Ha(a), Da(c)$
- $A_0(a, a), B_0(a, a), Agent(a), Agent(a), Agent(c), Ha(a), Ha(a), Da(c)$
- $A_1(a, a, n), B_0(a, a), I(\langle\{n\}_{pub(a)}, a\rangle), Agent(a), Agent(a), Agent(c), \ldots$
- $A_1(a, a, n), B_0(a, a), I(\langle\{n\}_{pub(a)}, a\rangle), I(c), I(\langle\{n\}_{pub(a)}, c\rangle), \ldots$
- $A_1(a, a, n), B_0(a, a), I(\langle\{n\}_{pub(a)}, c\rangle), B_0(c, a), A_0(c, a), \ldots$
- $A_1(a, a, n), B_0(a, a), B_1(c, a, n), I(\{n\}_{pub(c)}), \ldots$
- $A_1(a, a, n), B_0(a, a), B_1(c, a, n), I(\{n\}_{pub(c)}), I(n), I(\{n\}_{pub(a)}), \ldots$
- $A_2(a, a, n), B_0(a, a), B_1(c, a, n), I(\{n\}_{pub(c)}), I(n), I(\{n\}_{pub(a)}), \ldots$

- It is unrealistic to let an agent have a session with himself.
- However this assumption is used by many tools for analyzing protocols, and also by many models of cryptographic protocols.
- If agents are not allowed to have sessions with themselves then the number of agents required for detecting an attack depends on the number of agent variables appearing in the description of the security property.

Allowing agents to speak to themselves leads to new attacks.

A single-step toy protocol:

$$A \rightarrow B : \{A, B, Na\}_{K_b}, \{secret\}_{\{A,A,Na\}_{K_b}}$$

The secret is revealed if and only if A can have a session with himself.

Disallowing agents to speak to themselves

For every pair of distinct agents a and b we add a rule

$$\rightsquigarrow Distinct(a, b)$$

For every pair of dishonest agents d_1 and d_2 we add a rule

$$\rightsquigarrow Distinct(d_1, d_2)$$

(A dishonest agent can still have a session with himself)

We modify protocol dependent clauses by putting $Distinct$ predicates in the left hand side.

$$Agent(x), Agent(y), Distinct(x, y)$$
$$\rightsquigarrow A_0(x, y), Agent(x), Agent(y), Distinct(x, y)$$
$$\ldots$$

Reduction to $k + 1$ agents

Let the security property be of the form

$$Ha(x_1), \ldots, Ha(x_k), M$$

where Ha does not occur in M.

Suppose there is an attack in which the variables $x_1, \ldots, x_k$ get the values $h_1, \ldots, h_k$.

Fix one dishonest agent d.

We project all agents other than $h_1, \ldots, h_k$ to the dishonest agent d.

Define

$$\begin{aligned}
\texttt{proj}(h_i) &= h_i && 1 \leq i \leq k \\
\texttt{proj}(a) &= d && \text{other agents } a \\
\texttt{proj}(\{m_1\}_{m_2}) &= \{\texttt{proj}(m_1)\}_{\texttt{proj}(m_2)} && \\
\texttt{proj}(pub(a)) &= pub(\texttt{proj}(a)) && \\
\texttt{proj}(Ha(h_i)) &= Ha(h_i) && 1 \leq i \leq k \\
\texttt{proj}(Ha(a)) &= Da(d) && \text{other agents } a \\
\texttt{proj}(P(t_1, t_2, \ldots)) &= P(\texttt{proj}(t_1), \texttt{proj}(t_2), \ldots) && P \neq Ha \\
& \ldots && \\
\texttt{proj}(Distinct(a, b)) &= Distinct(\texttt{proj}(a), \texttt{proj}(b)) &&
\end{aligned}$$

‘$Distinct$’ agents remain distinct after projection.

An agent is required to be honest only in the rule

$$Ha(x) \leadsto Agent(x), Ha(x)$$

But we also have the rule

$$Da(x) \leadsto Agent(x), Da(x)$$

Hence as before, if a protocol state $F_1, \ldots, F_n$ is reachable then the protocol state $\texttt{proj}(F_1), \ldots, \texttt{proj}(F_n)$ is also reachable.

Hence $k + 1$ agents are sufficient for detecting an attack.