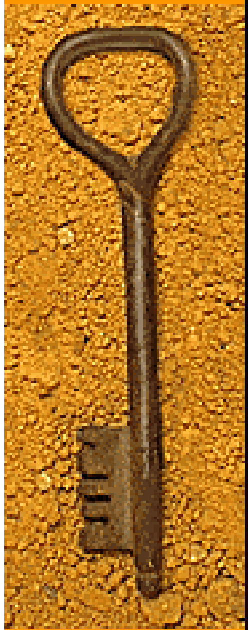
Formal Verification of Cryptographic Protocols in Spi-Calculus

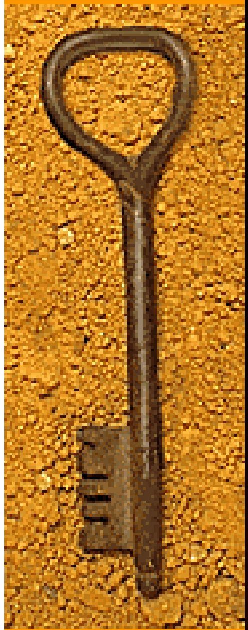
Eijiro Sumii

Tohoku University



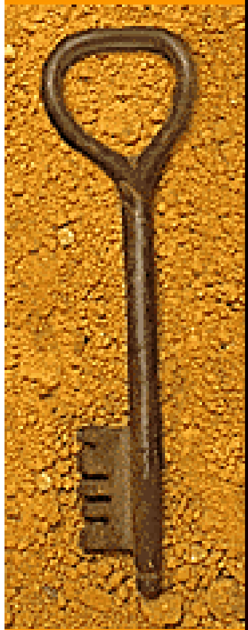
Caution

- Literature on spi-calculus is confusing
  - Inconsistent terminology
  - Some "results" found too weak or even wrong
- This talk is my own combination of various results on spi-calculus



Outline

- What is spi-calculus?
  – Syntax and operational semantics
- Example protocol
- Attack against the example protocol
- Formalizing secrecy by non-interference
- Proving secrecy by hedged bisimulations
- Conclusions



What is spi-calculus? [Abadi-Gordon 99]

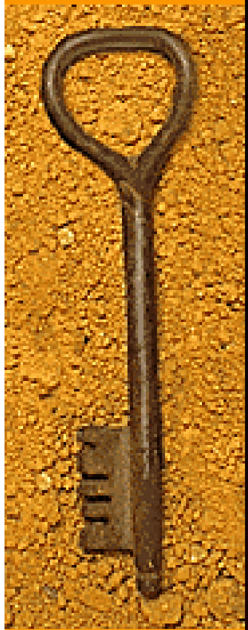
- spi-calulus = π-calculus + (shared-key) perfect encryption primitives

The only equation is:

dec(enc(Msg, key), key) = Msg

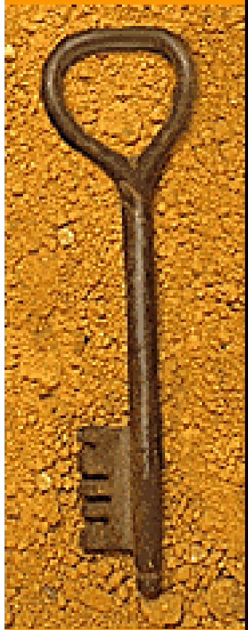
Cf. Textbook RSA is malleable:

$\mathrm{enc}(\mathrm{Msg}_1, \mathrm{pubkey}) \times \mathrm{enc}(\mathrm{Msg}_2, \mathrm{pubkey})$
$= \mathrm{enc}(\mathrm{Msg}_1 \times \mathrm{Msg}_2, \mathrm{pubkey})$



Syntax

$M, N ::=$	message
$\quad x$	name
$\quad \{M_1, \ldots, M_n\}_N$	ciphertext
$P, Q, R ::=$	process
$\quad 0$	inaction
$\quad \overline{M}\langle N\rangle.P$	sending
$\quad M(x).P$	receiving
$\quad P \mid Q$	parallel composition
$\quad (\nu x)P$	restriction
$\quad !P$	replication
$\quad \texttt{case}\ M\ \texttt{of}\ \{x_1, \ldots, x_n\}_N\ \texttt{in}\ P$	decryption
$\quad [M = N]P$	matching



Operational Semantics (1/2): Structural Equivalence

$$\texttt{case}\ \{M_1,\ldots,M_n\}_N\ \texttt{of}\ \{x_1,\ldots,x_n\}_N\ \texttt{in}\ P \equiv [M_1,\ldots,M_n/x_1,\ldots,x_n]P$$

$$[M = M]P \equiv P \qquad !P \equiv P \mid !P$$

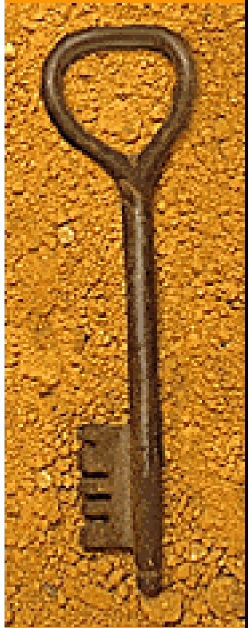
$$P \mid (\nu x)Q \equiv (\nu x)(P \mid Q) \quad \text{if } x \notin \mathit{free}(P)$$

$$P \mid 0 \equiv P \qquad P \mid Q \equiv Q \mid P \qquad (P \mid Q) \mid R \equiv P \mid (Q \mid R)$$

$$\frac{P \equiv P'}{P \mid Q \equiv P' \mid Q} \qquad \frac{P \equiv P'}{(\nu x)P \equiv (\nu x)P'}$$

$$P \equiv P \qquad \frac{P \equiv Q}{Q \equiv P} \qquad \frac{P \equiv Q \quad Q \equiv R}{P \equiv R}$$

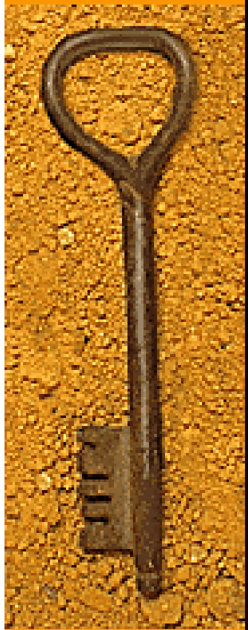
Operational Semantics (2/2): Reaction Relation



$$\overline{x}\langle M\rangle.P \mid x(y).Q \rightarrow P \mid [M/y]Q$$

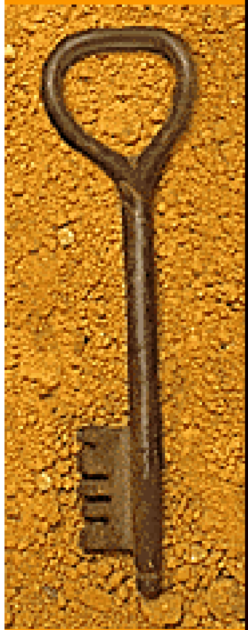
$$\frac{P \equiv P' \quad P' \rightarrow Q' \quad Q' \equiv Q}{P \rightarrow Q}$$

$$\frac{P \rightarrow P'}{P \mid Q \rightarrow P' \mid Q} \qquad \frac{P \rightarrow P'}{(\nu x)P \rightarrow (\nu x)P'}$$



Outline

- What is spi-calculus?
  - Syntax and operational semantics
- Example protocol
- Attack against the example protocol
- Formalizing secrecy by non-interference
- Proving secrecy by hedged bisimulations
- Conclusions



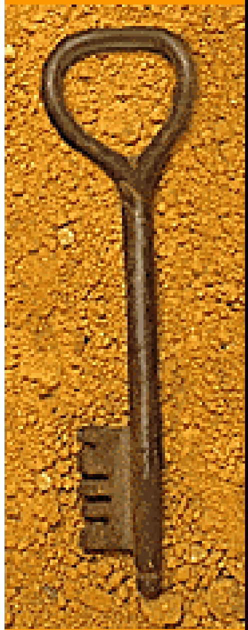
Example: A Naive Protocol (Wide Mouthed Frog Protocol)

$$\begin{array}{llcll}
1. & A & \rightarrow & S & : \{K_{AB}\}_{K_{AS}} \\
2. & S & \rightarrow & B & : \{K_{AB}\}_{K_{BS}} \\
3. & B & \rightarrow & A & : \{M\}_{K_{AB}}
\end{array}$$

$$\begin{array}{lcl}
P_A & = & (\nu K_{AB})\overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle. \\
 & & c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \\
P_S & = & c_{AS}(x).\texttt{case}\ x\ \texttt{of}\ \{y\}_{K_{AS}}\ \texttt{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \\
P_B & = & c_{BS}(x).\texttt{case}\ x\ \texttt{of}\ \{y\}_{K_{BS}}\ \texttt{in}\ \overline{c_{AB}}\langle\{M\}_y\rangle
\end{array}$$

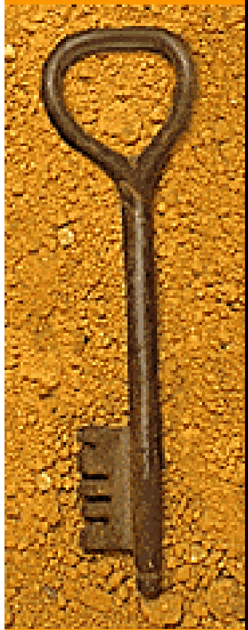
The whole system is:

$$(\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B)$$



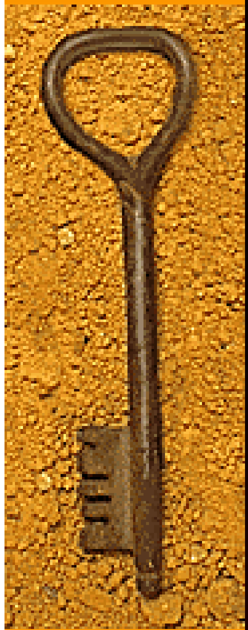
How does the protocol run? (1/2)

$$
\begin{array}{ll}
 & (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\
\equiv & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad (\overline{c_{AS}}\langle \{K_{AB}\}_{K_{AS}} \rangle . c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \mid \\
 & \quad c_{AS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{AS}}\ \mathtt{in}\ \overline{c_{BS}}\langle \{y\}_{K_{BS}} \rangle \mid \\
 & \quad c_{BS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{AB}}\langle \{M\}_y \rangle) \\
\rightarrow & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad (c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \mid \\
 & \quad \mathtt{case}\ \{K_{AB}\}_{K_{AS}}\ \mathtt{of}\ \{y\}_{K_{AS}}\ \mathtt{in}\ \overline{c_{BS}}\langle \{y\}_{K_{BS}} \rangle \mid \\
 & \quad c_{BS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{AB}}\langle \{M\}_y \rangle) \\
\equiv & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad (c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \mid \\
 & \quad \overline{c_{BS}}\langle \{K_{AB}\}_{K_{BS}} \rangle \mid \\
 & \quad c_{BS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{AB}}\langle \{M\}_y \rangle)
\end{array}
$$



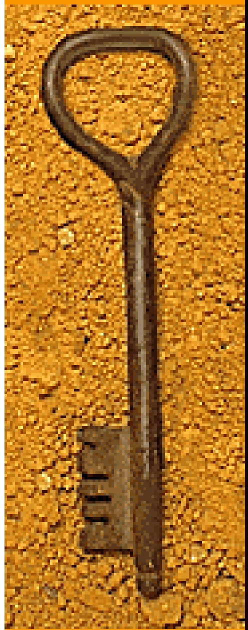
How does the protocol run? (2/2)

$$
\begin{array}{ll}
& (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
& \quad (c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \mid \\
& \quad \overline{c_{BS}}\langle \{K_{AB}\}_{K_{BS}} \rangle \mid \\
& \quad c_{BS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{AB}}\langle \{M\}_y \rangle) \\
\rightarrow & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
& \quad (c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \mid \\
& \quad \mathtt{case}\ \{K_{AB}\}_{K_{BS}}\ \mathtt{of}\ \{y\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{AB}}\langle \{M\}_y \rangle) \\
\equiv & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
& \quad (c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \mid \\
& \quad \overline{c_{AB}}\langle \{M\}_{K_{AB}} \rangle) \\
\rightarrow & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
& \quad \mathtt{case}\ \{M\}_{K_{AB}}\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \\
\equiv & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})0
\end{array}
$$



How does the protocol run? (2/2)

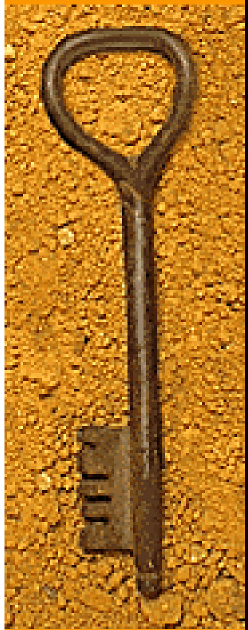
$$
\begin{array}{ll}
 & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad (c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \mid \\
 & \quad \overline{c_{BS}}\langle\{K_{AB}\}_{K_{BS}}\rangle \mid \\
 & \quad c_{BS}(x).\texttt{case}\ x\ \texttt{of}\ \{y\}_{K_{BS}}\ \texttt{in}\ \overline{c_{AB}}\langle\{M\}_y\rangle) \\
\rightarrow & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad (c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \mid \\
 & \quad \texttt{case}\ \{K_{AB}\}_{K_{BS}}\ \texttt{of}\ \{y\}_{K_{BS}}\ \texttt{in}\ \overline{c_{AB}}\langle\{M\}_y\rangle) \\
\equiv & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad (c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \mid \\
 & \quad \overline{c_{AB}}\langle\{M\}_{K_{AB}}\rangle) \\
\rightarrow & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB}) \\
 & \quad \texttt{case}\ \{M\}_{K_{AB}}\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \\
\equiv & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})0
\end{array}
$$



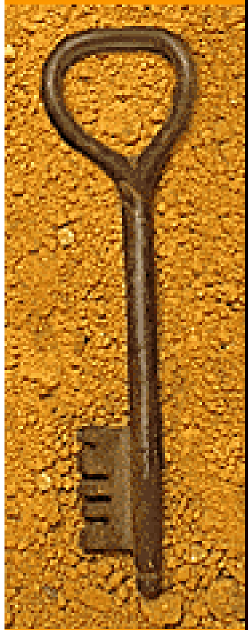
Outline

- What is spi-calculus?
  - Syntax and operational semantics
- Example protocol
- Attack against the example protocol
- Formalizing secrecy by non-interference
- Proving secrecy by hedged bisimulations
- Conclusions

Parallel runs of the protocol (1/2)



$$
\begin{array}{lllll}
1. & A & \rightarrow & S & : \{K_{AB}\}_{K_{AS}} \\
2. & S & \rightarrow & B & : \{K_{AB}\}_{K_{BS}} \\
3. & B & \rightarrow & A & : \{M\}_{K_{AB}} \\
 & & & & \\
1'. & B & \rightarrow & S & : \{K_{BE}\}_{K_{BS}} \\
2'. & S & \rightarrow & E & : \{K_{BE}\}_{K_{ES}} \\
3'. & E & \rightarrow & B & : \{M'\}_{K_{BE}}
\end{array}
$$



Parallel runs of the protocol (2/2)

$$P_A = (\nu K_{AB})\overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.\; c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0$$

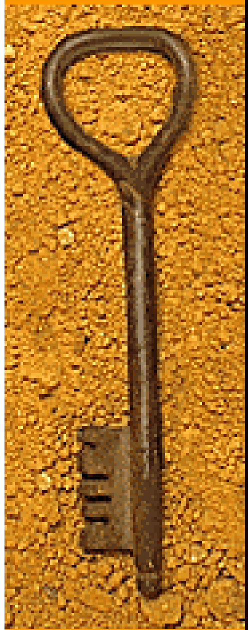
$$P_S = c_{AS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{AS}}\ \mathtt{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle$$

$$\mid\; c'_{BS}(x').\mathtt{case}\ x'\ \mathtt{of}\ \{y'\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle$$

$$P_B = c_{BS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y\}_{K_{BS}}\ \mathtt{in}\ \overline{c_{AB}}\langle\{M\}_y\rangle$$

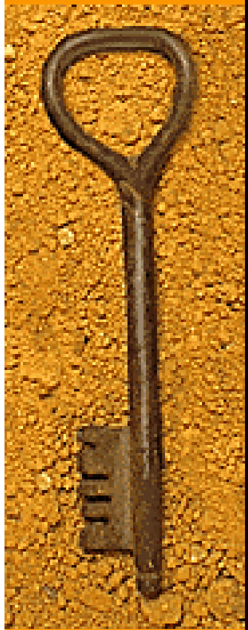
$$\mid\; (\nu K_{BE})\overline{c'_{BS}}\langle\{K_{BE}\}_{K_{BS}}\rangle.\; c_{BE}(n').\mathtt{case}\ n'\ \mathtt{of}\ \{m'\}_{K_{BE}}\ \mathtt{in}\ 0$$

$$P_E = c_{ES}(x').\mathtt{case}\ x'\ \mathtt{of}\ \{y'\}_{K_{ES}}\ \mathtt{in}\ \overline{c_{BE}}\langle\{M'\}_{y'}\rangle$$



Exercise (?)

- Write down the reduction of $(\nu K_{AS})(\nu K_{BS})(\nu K_{ES})(P_A \mid P_S \mid P_B \mid P_E)$.



What if E is evil in fact?

- Assumption: attacker has full access to open channels (Dolev-Yao model)
- Result: not only M' but also M may leak!

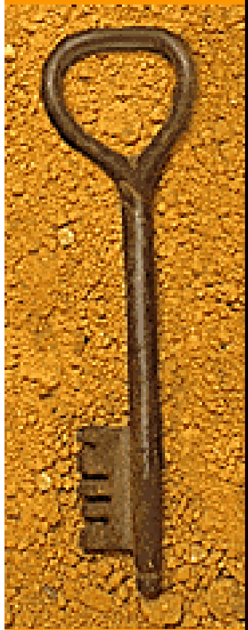
$$1'_a.\ B \rightarrow E(S)\ :\ \{K_{BE}\}_{K_{BS}}$$

$$2.\ E(S) \rightarrow B\ :\ \{K_{BE}\}_{K_{BS}}$$

$$1'_b.\ E(B) \rightarrow S\ :\ \{K_{BE}\}_{K_{BS}}$$

$$2'.\ S \rightarrow E\ :\ \{K_{BE}\}_{K_{ES}}$$

$$3.\ B \rightarrow E(A)\ :\ \{M\}_{K_{BE}}$$

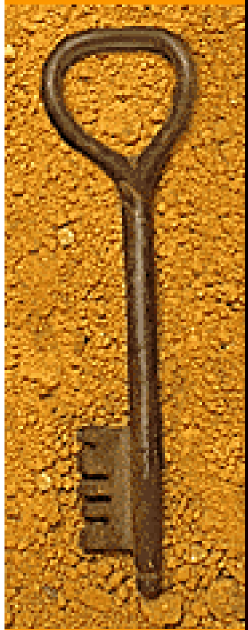


How does the attack work?

$$
\begin{aligned}
P'_E \;=\;& c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.\overline{c'_{BS}}\langle z\rangle.\\
& c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in}\\
& c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m
\end{aligned}
$$

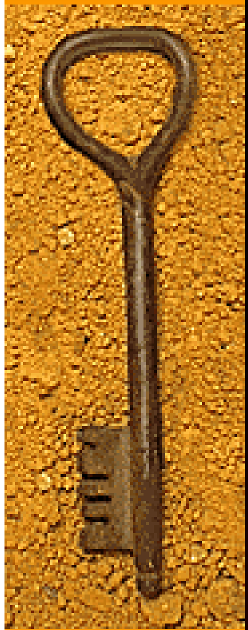
$$
\begin{aligned}
& P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B)\\
\equiv\; & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE})\\
& \quad (c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.\overline{c'_{BS}}\langle z\rangle.\\
& \quad c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in}\\
& \quad c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \mid\\
& \quad \overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid\\
& \quad c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid\\
& \quad c'_{BS}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle \mid\\
& \quad c_{BS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{AB}}\langle\{M\}_{y}\rangle \mid\\
& \quad \overline{c'_{BS}}\langle\{K_{BE}\}_{K_{BS}}\rangle.c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0)
\end{aligned}
$$

How does the attack work?



$$\begin{aligned} P'_E \;=\; & c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.c'_{BS}\langle z\rangle. \\ & c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\ & c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \end{aligned}$$

$$\begin{aligned} & P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\ \rightarrow\ & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\ & \quad (\overline{c_{BS}}\langle\{K_{BE}\}_{K_{BS}}\rangle.c'_{BS}\langle\{K_{BE}\}_{K_{BS}}\rangle. \\ & \quad c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\ & \quad c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \mid \\ & \quad \overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid \\ & \quad c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid \\ & \quad c'_{BS}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle \mid \\ & \quad c_{BS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{AB}}\langle\{M\}_y\rangle \mid \\ & \quad c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0) \end{aligned}$$



How does the attack work?

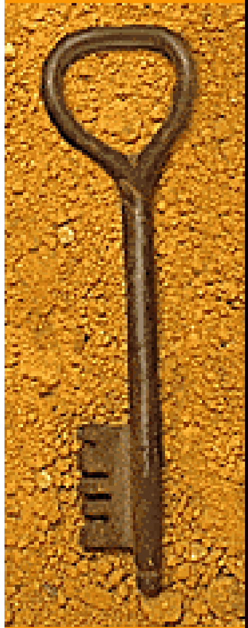
$$P'_E \;=\; c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.c'_{BS}\langle z\rangle.$$
$$c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in}$$
$$c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m$$

$$P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B)$$
$$\rightarrow\quad (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE})$$
$$(\overline{c_{BS}}\langle\{K_{BE}\}_{K_{BS}}\rangle.c'_{BS}\langle\{K_{BE}\}_{K_{BS}}\rangle.$$
$$c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in}$$
$$c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \mid$$
$$\overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid$$
$$c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid$$
$$c'_{BS}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle \mid$$
$$c_{BS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{AB}}\langle\{M\}_y\rangle \mid$$
$$c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0)$$

How does the attack work?

$$
\begin{aligned}
P'_E \;=\; & c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.c'_{BS}\langle z\rangle. \\
& c_{ES}(x').\texttt{case}\ x'\ \texttt{of}\ \{y'\}_{K_{ES}}\ \texttt{in} \\
& c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{y'}\ \texttt{in}\ \mathsf{DoEvil}_m
\end{aligned}
$$

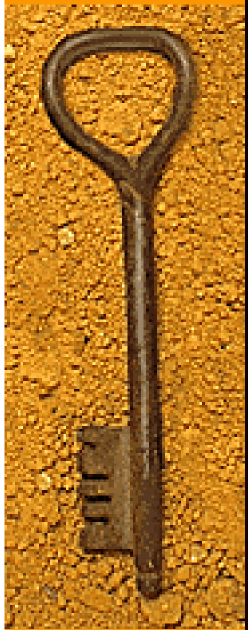
$$
\begin{aligned}
& P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\
\rightarrow^* \; & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\
& \quad (c'_{BS}\langle\{K_{BE}\}_{K_{BS}}\rangle. \\
& \quad c_{ES}(x').\texttt{case}\ x'\ \texttt{of}\ \{y'\}_{K_{ES}}\ \texttt{in} \\
& \quad c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{y'}\ \texttt{in}\ \mathsf{DoEvil}_m \mid \\
& \quad \overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \mid \\
& \quad c_{AS}(x).\texttt{case}\ x\ \texttt{of}\ \{y\}_{K_{AS}}\ \texttt{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid \\
& \quad c'_{BS}(x').\texttt{case}\ x'\ \texttt{of}\ \{y'\}_{K_{BS}}\ \texttt{in}\ \overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle \mid \\
& \quad \overline{c_{AB}}\langle\{M\}_{K_{BE}}\rangle \mid \\
& \quad c_{BE}(n').\texttt{case}\ n'\ \texttt{of}\ \{m'\}_{K_{BE}}\ \texttt{in}\ 0)
\end{aligned}
$$



How does the attack work?

$$\begin{array}{rcl} P'_E & = & c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.c'_{BS}\langle z\rangle. \\ & & c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\ & & c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \end{array}$$

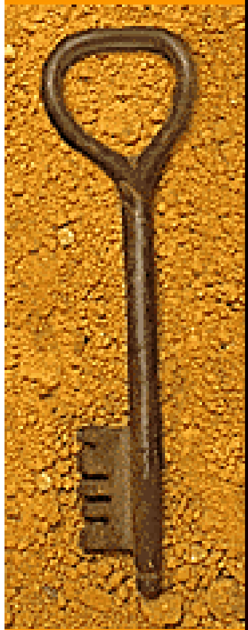
$$\begin{array}{rl} & P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\ \to^* & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\ & \quad (\overline{c'_{BS}}\langle\{K_{BE}\}_{K_{BS}}\rangle. \\ & \quad c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\ & \quad c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \mid \\ & \quad \overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid \\ & \quad c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid \\ & \quad c'_{BS}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{BS}}\ \mathsf{in}\ \overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle \mid \\ & \quad \overline{c_{AB}}\langle\{M\}_{K_{BE}}\rangle \mid \\ & \quad c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0) \end{array}$$



How does the attack work?

$$
\begin{aligned}
P'_E \;=\; & c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.\overline{c'_{BS}}\langle z\rangle. \\
& c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\
& c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m
\end{aligned}
$$

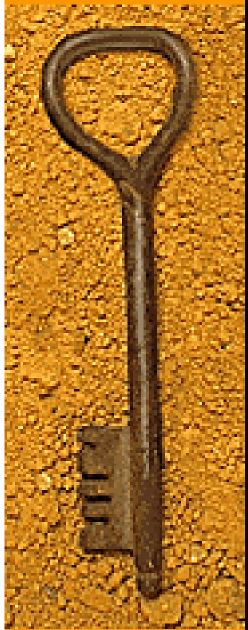
$$
\begin{aligned}
& P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\
\rightarrow^* \; & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\
& \quad (c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\
& \quad c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m \mid \\
& \quad \overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid \\
& \quad c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid \\
& \quad \overline{c_{ES}}\langle\{K_{BE}\}_{K_{ES}}\rangle \mid \\
& \quad \overline{c_{AB}}\langle\{M\}_{K_{BE}}\rangle \mid \\
& \quad c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0)
\end{aligned}
$$



How does the attack work?

$$
\begin{aligned}
P'_E \;=\;& c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.\overline{c'_{BS}}\langle z\rangle. \\
& c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\
& c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m
\end{aligned}
$$

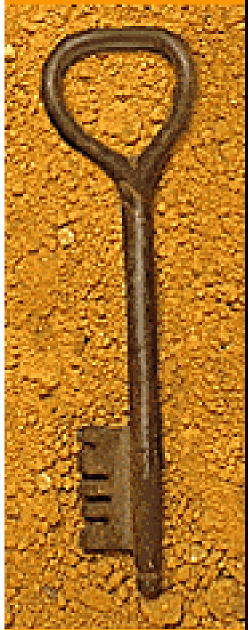
$$
\begin{aligned}
& P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\
\rightarrow^* \;& (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\
& \quad (c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{BE}}\ \mathsf{in}\ \mathsf{DoEvil}_m \mid \\
& \quad \overline{c_{AS}}\langle\{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid \\
& \quad c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \mid \\
& \quad \overline{c_{AB}}\langle\{M\}_{K_{BE}}\rangle \mid \\
& \quad c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0)
\end{aligned}
$$



How does the attack work?

$$
\begin{aligned}
P'_E \;=\;\; & c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.\overline{c'_{BS}}\langle z\rangle. \\
& c_{ES}(x').\texttt{case}\ x'\ \texttt{of}\ \{y'\}_{K_{ES}}\ \texttt{in} \\
& c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{y'}\ \texttt{in}\ \mathsf{DoEvil}_m
\end{aligned}
$$

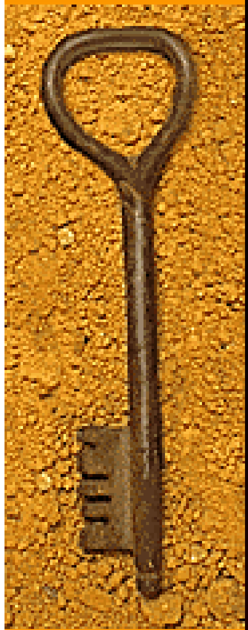
$$
\begin{aligned}
& P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\
\rightarrow^* \;& (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\
& \quad (c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{BE}}\ \texttt{in}\ \mathsf{DoEvil}_m \mid \\
& \quad \overline{c_{AS}}\langle \{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\texttt{case}\ n\ \texttt{of}\ \{m\}_{K_{AB}}\ \texttt{in}\ 0 \mid \\
& \quad c_{AS}(x).\texttt{case}\ x\ \texttt{of}\ \{y\}_{K_{AS}}\ \texttt{in}\ \overline{c_{BS}}\langle \{y\}_{K_{BS}}\rangle \mid \\
& \quad \overline{c_{AB}}\langle \{M\}_{K_{BE}}\rangle \mid \\
& \quad c_{BE}(n').\texttt{case}\ n'\ \texttt{of}\ \{m'\}_{K_{BE}}\ \texttt{in}\ 0)
\end{aligned}
$$



How does the attack work?

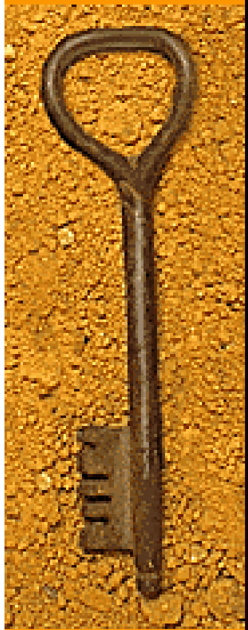
$$
\begin{aligned}
P'_E \;=\; & c'_{BS}(z).\overline{c_{BS}}\langle z\rangle.\overline{c'_{BS}}\langle z\rangle. \\
& c_{ES}(x').\mathsf{case}\ x'\ \mathsf{of}\ \{y'\}_{K_{ES}}\ \mathsf{in} \\
& c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{y'}\ \mathsf{in}\ \mathsf{DoEvil}_m
\end{aligned}
$$

$$
\begin{aligned}
& P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B) \\
\rightarrow^* \; & (\nu K_{AS})(\nu K_{BS})(\nu K_{AB})(\nu K_{BE}) \\
& \quad (\mathsf{DoEvil}_M \mid \\
& \quad \overline{c_{AS}}\langle \{K_{AB}\}_{K_{AS}}\rangle.c_{AB}(n).\mathsf{case}\ n\ \mathsf{of}\ \{m\}_{K_{AB}}\ \mathsf{in}\ 0 \mid \\
& \quad c_{AS}(x).\mathsf{case}\ x\ \mathsf{of}\ \{y\}_{K_{AS}}\ \mathsf{in}\ \overline{c_{BS}}\langle \{y\}_{K_{BS}}\rangle \mid \\
& \quad c_{BE}(n').\mathsf{case}\ n'\ \mathsf{of}\ \{m'\}_{K_{BE}}\ \mathsf{in}\ 0)
\end{aligned}
$$



Outline

- What is spi-calculus?
  - Syntax and operational semantics
- Example protocol
- Attack against the example protocol
- Formalizing secrecy by <u>non-interference</u>
- Proving secrecy by <u>hedged bisimulations</u>
- Conclusions



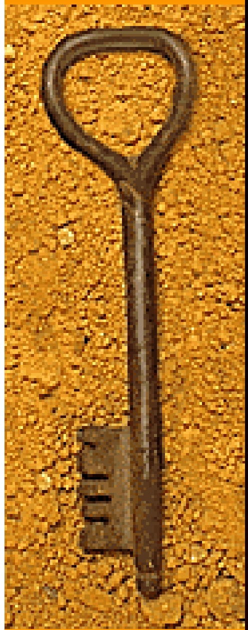
Formalizing secrecy by non-interference

- "Definition": Process P keeps message x totally secret if [M/x]P and [N/x]P are "equivalent" for any M and N

  Cf. partial secrecy: [M/x]P and [N/x]P are equivalent for any M and N satisfying some condition (e.g., M mod 2 = N mod 2)

- What equivalence should we take?

  ⇒ (Strong) barbed equivalence

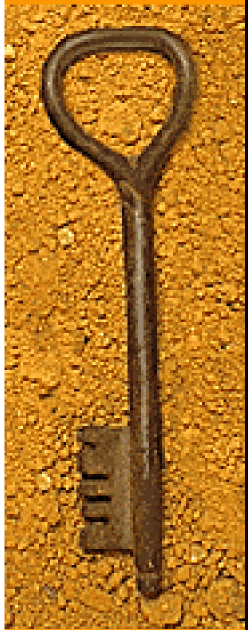


Definitions (1/2)

- Process P immediately exhibits input barb c, written $P \downarrow c$, if

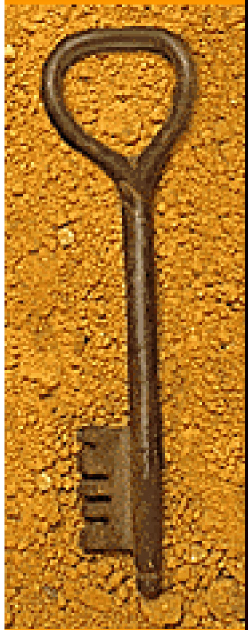
  $$P \equiv (\nu x_1)\ldots(\nu x_n)(c(y).Q \mid R)$$

  for some x_1, ..., x_n (distinct from c), y, Q and R. Similar for output.
- A (strong) barbed simulation S is a binary relation on processes such that $P\ S\ Q$ implies:
  - for each barb β, if $P \downarrow \beta$, then $Q \downarrow \beta$, and
  - if $P \to P'$, then $Q \to Q'$ and $P'\ S\ Q'$ for some Q
- S is a barbed bisimulation if both S and S^{-1} are barbed simulations



Definitions (2/2)

- Barbed bisimilarity is the largest barbed bisimulation
  - Equals the union of all barbed bisimulations, which is also a barbed bisimulation
- Processes P and Q are barbed equivalent if P | R and Q | R are barbed bisimilar for every R



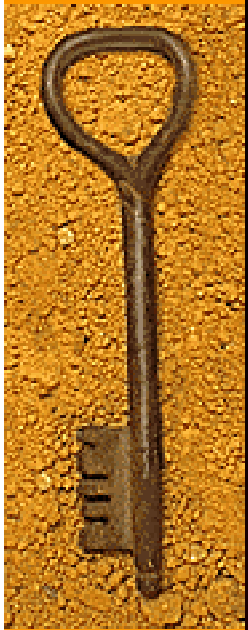
Example

- $(\nu k)\overline{c}\langle\{x\}_k\rangle$ keeps x totally secret. I.e., $(\nu k)\overline{c}\langle\{M\}_k\rangle$ and $(\nu k)\overline{c}\langle\{N\}_k\rangle$ are barbed equivalent for any M and N.

Proof sketch: given M and N, take

$$S = \{ (P, Q) \mid P \equiv (\nu k)\, [\{M\}_k/y]R, \; Q \equiv (\nu k)\, [\{N\}_k/y]R, \; k \notin free(R) \}$$

and prove it to be a barbed bisimulation by case analysis (and induction) on the reduction rules



Example

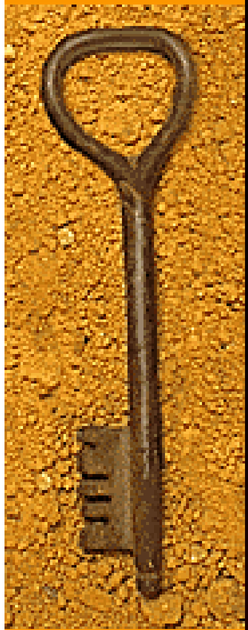
- $P = (\nu k)(\overline{c}\langle\{x\}_k\rangle \mid$
  $c(y).\mathtt{case}\ y\ \mathit{of}\ \{z\}_k\ \mathit{in}\ \overline{c}\langle k\rangle)$

  does not keep x totally secret. Indeed, [M/x]P and [N/x]P are not barbed equivalent for any M ≠ N.

Proof: given M and N, take

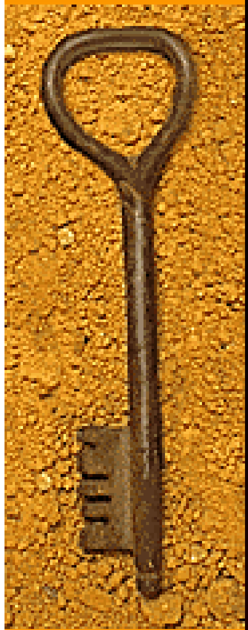
$$R \;=\; c(y).\overline{c}\langle y\rangle.c(k).\mathtt{case}\ y\ \mathtt{of}\ \{m\}_k\ \mathtt{in}\ [m = M]\overline{world}\langle hello\rangle$$

Cf. $P = (\nu k)(\overline{k}\langle x\rangle \mid k(y).\overline{c}\langle k\rangle)$ does keep x secret



Side Step: The Vice of May Testing Equivalence

- Many papers (including Abadi and Gordon's original work!) use <u>may testing equivalence</u> for defining secrecy by non-interference, but it is too weak

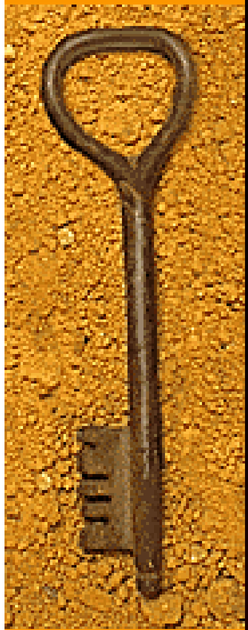


Definitions

- Process P <u>may eventually exhibit barb</u> β, written $P \Downarrow \beta$, if $P \rightarrow \ldots \rightarrow P' \downarrow \beta$ for some P'
- Processes P and Q are <u>may testing equivalent</u> if

  $$(P \mid R) \Downarrow \beta \iff (Q \mid R) \Downarrow \beta$$

  for every R and β



So what's wrong?

- Surprisingly,

$$P \;=\; (\nu d)(\overline{d}\langle\rangle \mid d().\overline{c}\langle\rangle)$$

  and

$$Q \;=\; (\nu d)(\overline{d}\langle\rangle \mid d().\overline{c}\langle\rangle \mid d().0)$$

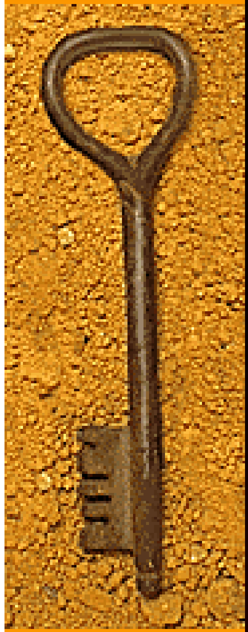
  are may testing equivalent.

- As a result, processes like

  if x > 0 then P else Q

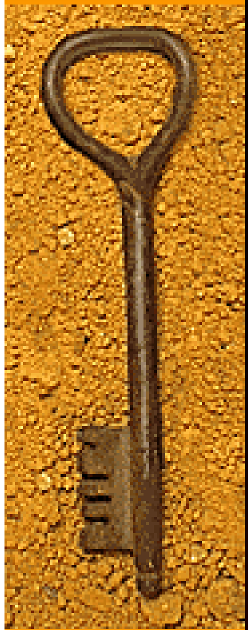
  are regarded as keeping x totally secret (under may testing equivalence)

- But the leak is possible!



Outline

- What is spi-calculus?
  - Syntax and operational semantics
- Example protocol
- Attack against the example protocol
- Formalizing secrecy by non-interference
- Proving secrecy by hedged bisimulations
- Conclusions

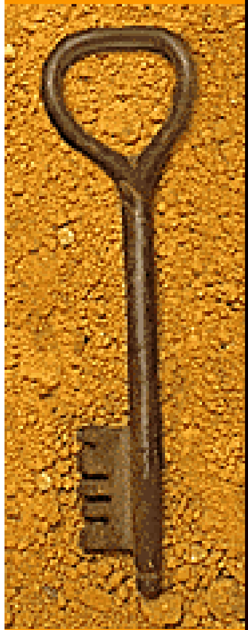


Hedged Bisimulation: Motivation

Direct proof of barbed equivalence is difficult because of "arbitrary R"

⇒ Devise a proof technique without "arbitrary R"

- What can R do?
  - Gain "knowledge" by receiving from a known channel
  - Send to a known channel a message <u>synthesized</u> from the knowledge

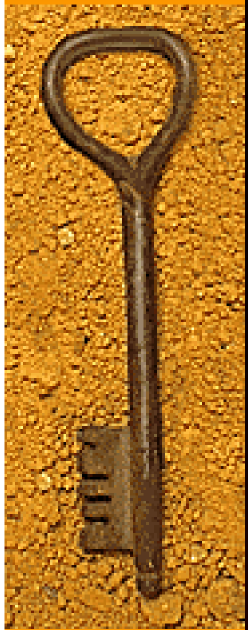


Definitions (1/4)

- A hedge H is a binary relation on messages
- $H \vdash M \leftrightarrow N$ (messages M and N can be synthesized from hedge H) is defined by induction:

$$\frac{(M,N) \in \mathcal{H}}{\mathcal{H} \vdash M \leftrightarrow N} \qquad \frac{\mathcal{H} \vdash M_1 \leftrightarrow N_1 \quad \mathcal{H} \vdash M_2 \leftrightarrow N_2}{\mathcal{H} \vdash \{M_1\}_{M_2} \leftrightarrow \{N_1\}_{N_2}}$$

$$\frac{\mathcal{H} \vdash \{M_1\}_{M_2} \leftrightarrow \{N_1\}_{N_2} \quad \mathcal{H} \vdash M_2 \leftrightarrow N_2}{\mathcal{H} \vdash M_1 \leftrightarrow N_1} \qquad \frac{x \notin \mathit{free}(\mathcal{H})}{\mathcal{H} \vdash x \leftrightarrow x}$$



Definitions (2/4)

- A hedged simulation is a set X of triples (P, Q, H) that satisfies:

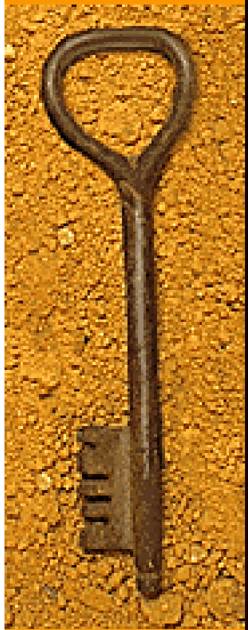
1. For any $P \to P'$, there exists some Q' such that $Q \to Q'$ and $(P', Q', \mathcal{H}) \in X$.
2. If for some $\mathcal{H} \vdash c \leftrightarrow d$,

$$P \equiv (\nu x_1) \ldots (\nu x_m)(\overline{c}\langle M \rangle.P_1 \mid P_2) \qquad x_i \notin \{c\} \cup \mathit{free}(\mathit{fst}(\mathcal{H})),$$

then

$$Q \equiv (\nu y_1) \ldots (\nu y_n)(\overline{d}\langle N \rangle.Q_1 \mid Q_2) \qquad y_i \notin \{d\} \cup \mathit{free}(\mathit{snd}(\mathcal{H}))$$

and $(P_1 \mid P_2, Q_1 \mid Q_2, \mathcal{H} \cup (M, N)) \in X$.



Definitions (3/4)

3. If for some $\mathcal{H} \vdash c \leftrightarrow d$,

$$P \equiv (\nu x_1) \ldots (\nu x_m)(c(z).P_1 \mid P_2)$$
$$x_i \notin \{c\} \cup \mathit{free}(\mathit{fst}(\mathcal{H})),$$

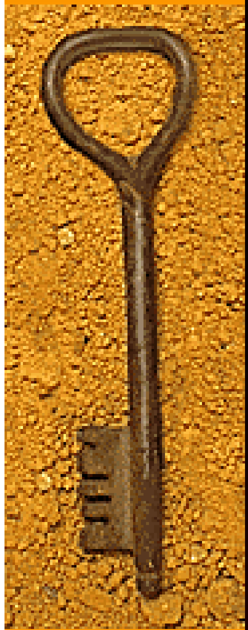
then $Q \equiv (\nu y_1) \ldots (\nu y_n)(d(z).Q_1 \mid Q_2)$

$$y_i \notin \{d\} \cup \mathit{free}(\mathit{snd}(\mathcal{H}))$$

and for any $\mathcal{H} \vdash M \leftrightarrow N$,

$$([M/z]P_1 \mid P_2, [N/z]Q_1 \mid Q_2, \mathcal{H}) \in X.$$

4. If $\mathcal{H} \vdash M_1 \leftrightarrow N_1$ and $\mathcal{H} \vdash M_2 \leftrightarrow N_2$, then $M_1 = M_2$ implies $N_1 = N_2$.
5. If $\mathcal{H} \vdash \{M_1\}_{M_2} \leftrightarrow N$ and $\mathcal{H} \vdash M_2 \leftrightarrow N_2$, then $N = \{N_1\}_{N_2}$ for some N_1.

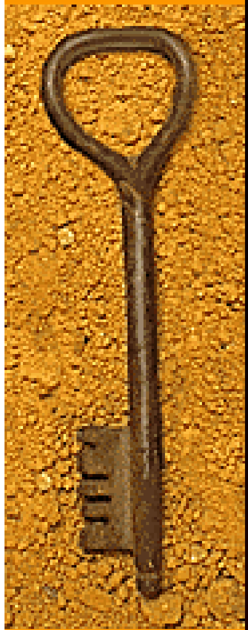


Definitions (4/4)

- A hedged simulation X is a hedged bisimulation if X^{-1} is also a hedged simulation, where X^{-1} is defined as:

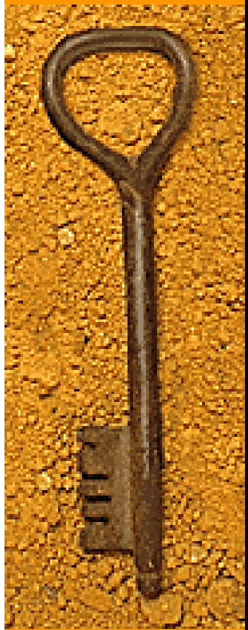
$$\{(Q, P, H^{-1}) \mid (P, Q, H) \in X\}$$

- Hedged bisimilarity is the largest hedged bisimulation (i.e., the union of all hedged bisimulations, which is also a hedged bisimulation)
- Notation: $P \sim_H Q \Leftrightarrow (P, Q, H)$ is in the hedged bisimilarity



Caution: α-Conversion of Hedged Bisimulation

- Every (P, Q, H) $\in$ X is regarded as α-equivalent to
  $(\sigma P, Q, \{ (\sigma M, N) \mid (M, N) \in H \})$
  for every $dom(\sigma) \supseteq free(P) \cup free(fst(H))$
- Every (P, Q, H) $\in$ X is regarded as α-equivalent to
  $(P, \sigma Q, \{ (M, \sigma N) \mid (M, N) \in H \})$
  for every $dom(\sigma) \supseteq free(Q) \cup free(snd(H))$
- Everything in the rest is considered "up to" this α-equivalence



Example 1

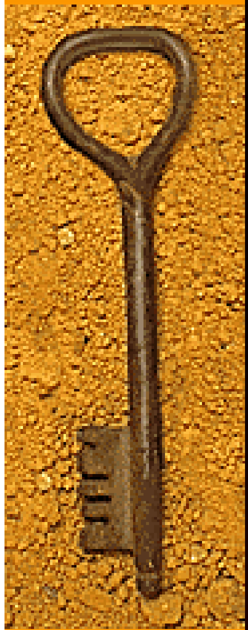
- For any M and N,

$$(\nu k)\overline{c}\langle\{M\}_k\rangle.0 \sim_{\{(c,c)\}} (\nu k)\overline{c}\langle\{N\}_k\rangle.0$$

Proof: take

$$\begin{array}{rcl} X & = & \{((\nu k)\overline{c}\langle\{M\}_k\rangle.0, \\ & & \quad (\nu k)\overline{c}\langle\{N\}_k\rangle.0, \\ & & \quad \{(c,c)\})\} \\ & \cup & \{(0, \\ & & \quad 0, \\ & & \quad \{(c,c),(\{M\}_k,\{N\}_k)\})\} \end{array}$$

and check conditions 1-5.

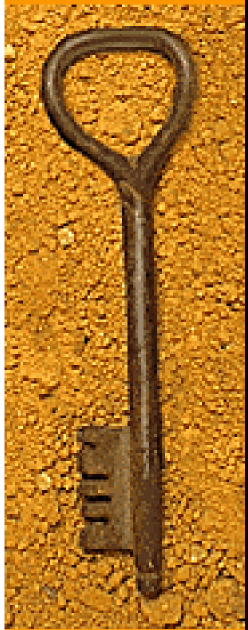


Example 2

- $(\nu k)(\nu n)\overline{c}\langle\{n\}_k\rangle.(\nu m)\overline{c}\langle m\rangle \sim_{\{(c,c)\}} (\nu k)(\nu n)\overline{c}\langle\{n\}_k\rangle.\overline{c}\langle n\rangle$

Proof: take

$$\begin{array}{rcl} X & = & \{((\nu k)(\nu n)\overline{c}\langle\{n\}_k\rangle.(\nu m)\overline{c}\langle m\rangle, \\ & & \quad (\nu k)(\nu n)\overline{c}\langle\{n\}_k\rangle.\overline{c}\langle n\rangle, \\ & & \quad \{(c,c)\})\} \\ & \cup & \{((\nu m)\overline{c}\langle m\rangle, \\ & & \quad \overline{c}\langle n\rangle, \\ & & \quad \{(c,c),(\{n\}_k,\{n\}_k)\})\} \\ & \cup & \{(0, \\ & & \quad 0, \\ & & \quad \{(c,c),(\{n\}_k,\{n\}_k),(m,n)\})\}. \end{array}$$

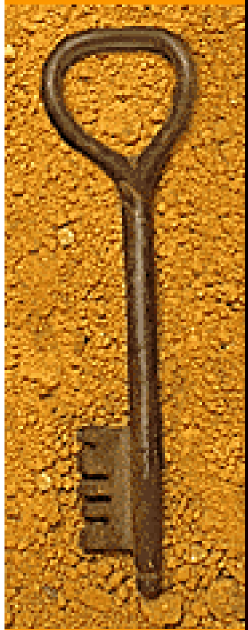


Example 3

- $(\nu k)(\nu n)(\nu l)\overline{c}\langle\{\{n\}_k\}_l\rangle.(\nu m)\overline{c}\langle m\rangle \sim_{\{(c,c)\}} (\nu k)(\nu n)\overline{c}\langle\{n\}_k\rangle.(\nu m)\overline{c}\langle m\rangle$

Proof: take

$$\begin{aligned}
X \quad = \quad & \{((\nu k)(\nu n)(\nu l)\overline{c}\langle\{\{n\}_k\}_l\rangle.(\nu m)\overline{c}\langle m\rangle, \\
& \quad (\nu k)(\nu n)\overline{c}\langle\{n\}_k\rangle.(\nu m)\overline{c}\langle m\rangle, \\
& \quad \{(c,c)\})\} \\
\cup \quad & \{((\nu m)\overline{c}\langle m\rangle, \\
& \quad (\nu m)\overline{c}\langle m\rangle, \\
& \quad \{(c,c),(\{\{n\}_k\}_l,\{n\}_k)\})\} \\
\cup \quad & \{(0, \\
& \quad 0, \\
& \quad \{(c,c),(\{\{n\}_k\}_l,\{n\}_k),(m,m)\})\}.
\end{aligned}$$



Theorem

Hedged bisimilarity is sound w.r.t. barbed equivalence. I.e., if $P \sim_H Q$ for

$$H = \{ (x, x) \mid x \in free(P) \cup free(Q) \},$$

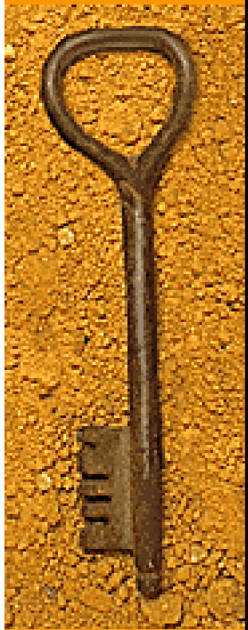
then P and Q are barbed equivalent.

Proof sketch: take

$$\begin{aligned}
S = \{ (P', Q') \mid\ & P \sim_H Q, \\
& P' \equiv (\nu x_1)\ldots(\nu x_l)\ (P \mid [M_1,\ldots,M_n/z_1,\ldots,z_n]R), \\
& Q' \equiv (\nu y_1)\ldots(\nu y_m)\ (Q \mid [N_1,\ldots,N_n/z_1,\ldots,z_n]R), \\
& H \vdash M_1 \leftrightarrow N_1, \ldots, H \vdash M_n \leftrightarrow N_n, \\
& free(R) \text{ distinct from } free(P), free(Q), \text{ and } free(H)) \}
\end{aligned}$$

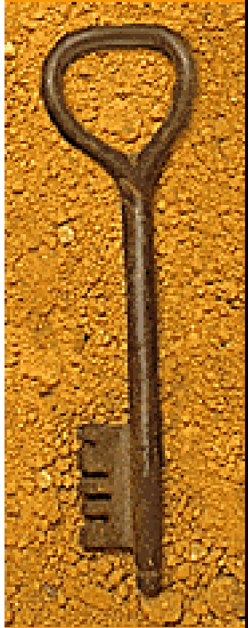
and prove it to be a barbed bisimulation by case analysis (and induction) on the reduction rules.

Real Example: Fixed Version of Previous Protocol



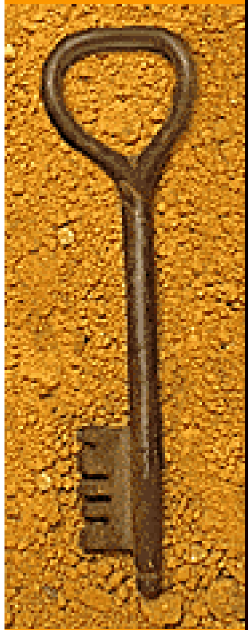
$$
\begin{array}{llllll}
1. & A & \rightarrow & S & : & \{K_{AB}, B\}_{K_{AS}} \\
2. & S & \rightarrow & B & : & \{K_{AB}, A\}_{K_{BS}} \\
3. & B & \rightarrow & A & : & \{M\}_{K_{AB}}
\end{array}
$$

$$
\begin{array}{llllll}
1'. & B & \rightarrow & S & : & \{K_{BE}, E\}_{K_{BS}} \\
2'. & S & \rightarrow & E & : & \{K_{BE}, B\}_{K_{ES}} \\
3'. & E & \rightarrow & B & : & \{M'\}_{K_{BE}}
\end{array}
$$



As Spi-Calculus Processes...

$$
\begin{array}{rcl}
P_A & = & (\nu K_{AB})\overline{c_{AS}}\langle\{K_{AB}, B\}_{K_{AS}}\rangle. \\
& & c_{AB}(n).\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0 \\
P_S & = & c_{AS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y, b\}_{K_{AS}}\ \mathtt{in} \\
& & [b = B]\overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle \\
& | & c'_{BS}(x').\mathtt{case}\ x'\ \mathtt{of}\ \{y', e\}_{K_{BS}}\ \mathtt{in} \\
& & [e = E]\overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle \\
P_B & = & c_{BS}(x).\mathtt{case}\ x\ \mathtt{of}\ \{y, a\}_{K_{BS}}\ \mathtt{in} \\
& & [a = A]\overline{c_{AB}}\langle\{z\}_y\rangle \\
& | & (\nu K_{BE})\overline{c'_{BS}}\langle\{K_{BE}, E\}_{K_{BS}}\rangle. \\
& & c_{BE}(n').\mathtt{case}\ n'\ \mathtt{of}\ \{m'\}_{K_{BE}}\ \mathtt{in}\ 0
\end{array}
$$

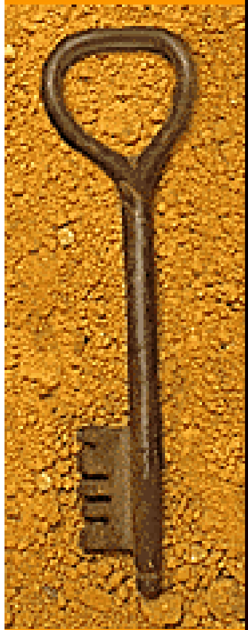


Exercise (?)

- Write down the reduction(s) of

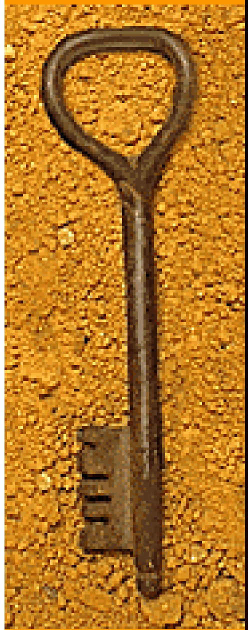
  $$P'_E \mid (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B)$$

  for the same attacker P'_E as before, for the fixed version of P_A, P_S, and P_B. Pinpoint where the attack fails.



Claim

- $(\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid P_B)$
  keeps z totally secret. I.e.,
  $P = (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid [M/z]P_B)$
  and
  $Q = (\nu K_{AS})(\nu K_{BS})(P_A \mid P_S \mid [N/z]P_B)$
  are barbed equivalent for any M and N.

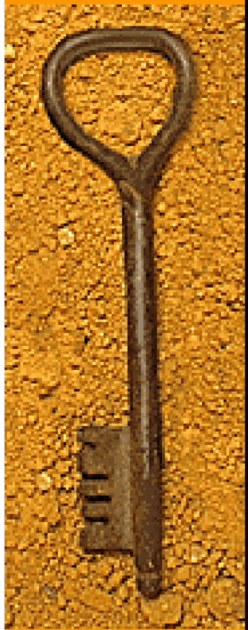


Proof Sketch

- Let $H = \{ (x, x) \mid x \in free(P) \cup free(Q) \}$
- We construct some hedged bisimulation $X \ni (P, Q, H)$
  - – The X is far from minimal, but this is fine as far as X is a hedged bisimulation
    - It is a nightmare to write down minimal X for real...

$$
\begin{aligned}
P_A &= (\nu K_{AB})\,\overbrace{\overline{c_{AS}}\langle\{K_{AB},B\}_{K_{AS}}\rangle.\,\overbrace{c_{AB}(n).\,\overbrace{\mathtt{case}\ n\ \mathtt{of}\ \{m\}_{K_{AB}}\ \mathtt{in}\ 0}^{P_{A_2}}}^{P_{A_1}}}^{P_{A_0}} \\
P_S &= \overbrace{c_{AS}(x).\,\overbrace{\mathtt{case}\ x\ \mathtt{of}\ \{y,b\}_{K_{AS}}\ \mathtt{in}\ [b=B]\,\overbrace{\overline{c_{BS}}\langle\{y\}_{K_{BS}}\rangle}^{P_{S_3}}}^{P_{S_1}}}^{P_{S_0}} \\
&\ \ \mid\ \underbrace{c'_{BS}(x').\,\underbrace{\mathtt{case}\ x'\ \mathtt{of}\ \{y',e\}_{K_{BS}}\ \mathtt{in}\ \underbrace{[e=E]\,\underbrace{\overline{c_{ES}}\langle\{y'\}_{K_{ES}}\rangle}_{P'_{S_3}}}_{P'_{S_2}}}_{P'_{S_1}}}_{P'_{S_0}}
\end{aligned}
$$

$$
\begin{aligned}
P_B \;&=\; \overbrace{c_{BS}(x).\,\overbrace{\texttt{case}\ x\ \texttt{of}\ \{y,a\}_{K_{BS}}\ \texttt{in}\ \overbrace{[a=A]\,\overbrace{\overline{c_{AB}}\langle\{z\}_y\rangle}^{P_{B_3}}}^{P_{B_2}}}^{P_{B_1}}}^{P_{B_0}} \\
&\;\mid\;\; \underbrace{(\nu K_{BE})\,\underbrace{\overline{c'_{BS}}\langle\{K_{BE},E\}_{K_{BS}}\rangle.c_{BE}(n').\,\underbrace{\texttt{case}\ n'\ \texttt{of}\ \{m'\}_{K_{BE}}\ \texttt{in}\ 0}_{P'_{B_2}}}_{P'_{B_1}}}_{P'_{B_0}}
\end{aligned}
$$



$X = \{ (P', Q', H') \mid$

$P' \equiv (\nu c_1)\ldots(\nu c_u)$

$\quad ([M_1/n]P_{A_i} \mid [M_2/x]P_{S_j} \mid [M_3,A/x',e]P'_{S_k} \mid$

$\quad [M_4,E,M/x,a,z]P_{B_l} \mid [M_5/n']P'_{B_m}),$

$Q' \equiv (\nu d_1)\ldots(\nu d_v)$

$\quad ([N_1/n]P_{A_i} \mid [N_2/x]P_{S_j} \mid [N_3,A/x',e]P'_{S_k} \mid$

$\quad [N_4,E,N/x,a,z]P_{B_l} \mid [N_5/n']P'_{B_m}),$

$H' \subseteq H \cup \{ (\{K_{AB},B\}_{K_{AS}}, \{K_{AB},B\}_{K_{AS}}),$

$\quad (\{K_{AB}, A\}_{K_{BS}}, \{K_{AB}, A\}_{K_{BS}}),$

$\quad (\{M\}_{K_{AB}}, \{N\}_{K_{AB}}),$

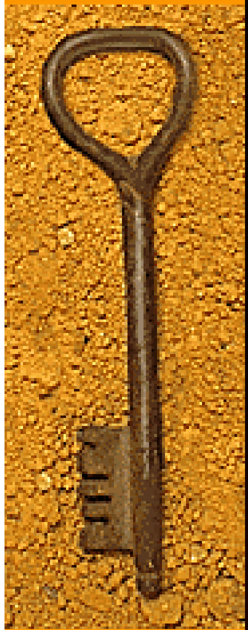
$\quad (\{K_{BE}, E\}_{K_{BS}}, \{K_{BE}, E\}_{K_{BS}}),$

$\quad (\{K_{BE}, B\}_{K_{ES}}, \{K_{BE}, B\}_{K_{ES}}) \},$

$H' \vdash M_w \leftrightarrow N_w$ for $w = 1, 2, 3, 4, 5,$

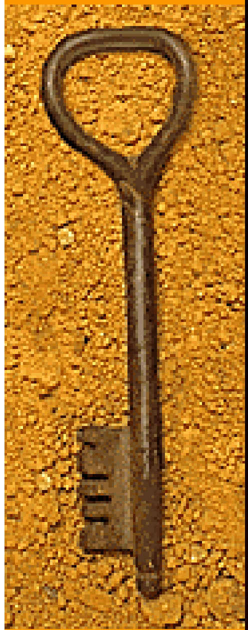
$c_1, \ldots, c_u \notin free(fst(H')),$

$d_1, \ldots, d_v \notin free(snd(H')) \}$



Exercise (?)

- Try to prove the total secrecy of z in the original version of this protocol by means of hedged bisimulation. Explain how the "proof" fails.



Side Step II: Completeness of Hedged Bisimulation

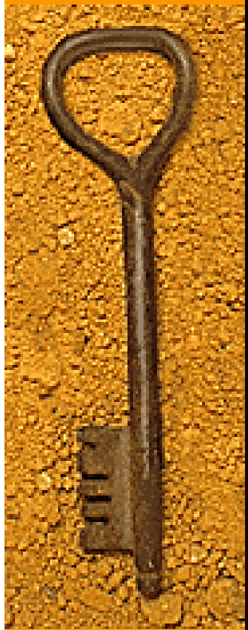
Conjecture:

Hedged bisimilarity is complete with respect to barbed equivalence.

I.e., if P and Q are barbed equivalent, then $P \sim_H Q$ for

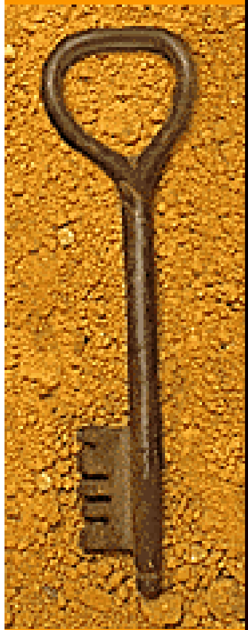
$$H = \{ (x, x) \mid x \in free(P) \cup free(Q) \}$$

– Proved for "structurally image finite" processes, but not for the general case (to my knowledge)



Outline

- What is spi-calculus?
  - Syntax and operational semantics
- Example protocol
- Attack against the example protocol
- Formalizing secrecy by non-interference
- Proving secrecy by hedged bisimulations
- Conclusions



Other Topics in Spi-Calculus

- Other bisimulations [Abadi-Gordon 98] [Boreale-DeNicola-Pugliese 99] [Elkjær-Höhle-Hüttel-Overgård 99]
  - More complex and "less complete"
- Secrecy by typing [Abadi 97] [Abadi-Blanchet 01]
- Authenticity by typing [Gordon-Jeffery 01] [Gordon-Jeffery 02] [Blanchet 02]

Cf. http://www.soe.ucsc.edu/~abadi/
http://www.di.ens.fr/~blanchet/
http://netlib.bell-labs.com/who/ajeffrey/ etc.