

ArmoredSoftware Semantics 0.0

ArmoredSoftware Crew
Information and Telecommunication Technology Center
The University of Kansas
palexand@ku.edu

December 5, 2014

Contents

1	Introduction	1
2	SPI Calculus	1
2.1	Wide Mouth Frog	2
2.2	Needham Schroeder	2
2.3	Privacy CA Protocol	5
3	Strand Spaces	5
3.1	Needham Schroder	5
A	Glossary	5

List of Figures

Abstract

This document describes evolving ARMOREDSOFTWARE semantic definitions.

1 Introduction

2 SPI Calculus

Examples motivated by Abadi and Gordon [1999].

2.1 Wide Mouth Frog

2.2 Needham Schroeder

$$\begin{aligned}
A &\rightarrow B : \{A^+, N_A\}_{B^+} \text{ on } c \\
B &\rightarrow A : \{N_A, N_B\}_{A^+} \text{ on } c \\
A &\rightarrow B : \{N_B\}_{B^+} \text{ on } c
\end{aligned}$$

$$\begin{aligned}
A &\triangleq \bar{c}\langle\{(A, N_A)\}_{B^+}\rangle. \\
&\quad c(M). \\
&\quad \text{case } \{M\}_{A^-} \text{ of } (N_A, nb) \text{ in} \\
&\quad \bar{c}\langle\{nb\}_{B^+}\rangle. \\
&\quad A \\
B &\triangleq c(M). \\
&\quad \text{case } \{M\}_{B^-} \text{ of } (x, n) \text{ in} \\
&\quad \bar{c}\langle\{(n, N_B)\}_{x^+}\rangle. \\
&\quad c(M). \\
&\quad \text{case } \{M\}_{B^-} \text{ of } N_B \text{ in } B \\
sys &\triangleq (\nu c)A \mid B
\end{aligned}$$

$$\text{React Inter} \frac{}{\bar{m}\langle N \rangle.P \mid m(x).Q \rightarrow P \mid [x \rightarrow N]Q}$$

$$\text{Red Replace} \frac{}{!P > P \mid !P}$$

$$\text{Red Match} \frac{}{\llbracket M \text{ is } M \rrbracket P > P}$$

$$\text{Red Let} \frac{}{\text{let } (x, y) = (M, N) \text{ in } P > [x \rightarrow M][y \rightarrow N]P}$$

Note that we may want a more general **let** that matches more than pairs here. We'll see what the other inference rules give us.

$$\text{Red Zero} \frac{}{\text{case } 0 \text{ of } 0 : P \text{ suc}(x) : Q > P}$$

$$\text{Red Suc} \frac{}{\text{case } \text{suc}(M) \text{ of } 0 : P \text{ suc}(x) : Q > [x \rightarrow M]Q}$$

I find the **case** rules over naturals quite crude.

$$\text{Red Sym Decrypt} \frac{}{\text{case } \{M\}_k \text{ of } \{x\}_k \text{ in } P > [x \rightarrow M]P}$$

Additional proposed semantic rules for public/private key encryption and signature checking

$$\text{Red Asym Decrypt 1} \frac{}{\text{case } \{M\}_{k^+} \text{ of } \{x\}_{k^-} \text{ in } P > [x \rightarrow M]P}$$

$$\text{Red Asym Decrypt 2} \frac{}{\text{case } \{M\}_{k^-} \text{ of } \{x\}_{k^+} \text{ in } P > [x \rightarrow M]P}$$

The previous two rules capture the essence of asymmetric key pairs. Specifically, encrypt with one and decrypt with the other.

Assume $|M|$ is the hash and not the message itself and $\{|M|\}_{k^-}$ is the hash encrypted with private key, k^- . Thus, a signed message is the pair $(M, \{|M|\}_{k^-})$ consisting of the message and the signed hash. Given this, a successful signature check looks something like this:

$$\begin{aligned} & \text{let } (m, s) = (M, \{|M|\}_{k^-}) \text{ in case } s \text{ of } \{x\}_{k^+} \text{ in } \llbracket x \text{ is } |m| \rrbracket P \\ & > [m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] \text{case } s \text{ of } \{x\}_{k^+} \text{ in } \llbracket x \text{ is } |m| \rrbracket P \\ & > \text{case } \{|M|\}_{k^-} \text{ of } \{x\}_{k^+} \text{ in } [m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] \llbracket x \text{ is } |m| \rrbracket P \\ & > [x \rightarrow |M|][m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] \llbracket x \text{ is } |m| \rrbracket P \\ & > \llbracket |M| \text{ is } |M| \rrbracket [x \rightarrow |M|][m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] P \\ & > [x \rightarrow |M|][m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] P \end{aligned}$$

This is precisely what we want. Specifically, P with m replaced by the message M and s replaced by the decrypted signature, $|M|$, produced by the signature check. It is unlikely that $|M|$ will be used in P , but it is available.

If the signature does not match, the process hangs. Assume the hash is incorrect. Specifically, $M \neq N$:

$$\begin{aligned} & \text{let } (m, s) = (M, \{|N|\}_{k^-}) \text{ in case } s \text{ of } \{x\}_{k^+} \text{ in } \llbracket x \text{ is } |m| \rrbracket P \\ & > [m \rightarrow M][s \rightarrow \{|N|\}_{k^-}] \text{case } s \text{ of } \{x\}_{k^+} \text{ in } \llbracket x \text{ is } |m| \rrbracket P \\ & > \text{case } \{|N|\}_{k^-} \text{ of } \{x\}_{k^+} \text{ in } [m \rightarrow M][s \rightarrow \{|N|\}_{k^-}] \llbracket x \text{ is } |m| \rrbracket P \\ & > [x \rightarrow |N|][m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] \llbracket x \text{ is } |m| \rrbracket P \\ & > \llbracket |N| \text{ is } |M| \rrbracket [x \rightarrow |N|][m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] P \end{aligned}$$

The process is stuck when $|N|$ is $|M|$ fails because $N \neq M$.

Now assume the wrong private key was used to sign the message hash. Specifically, $j \neq k$:

$$\begin{aligned} & \text{let } (m, s) = (M, \{|M|\}_{j^-}) \text{ in case } s \text{ of } \{x\}_{k^+} \text{ in } \llbracket x \text{ is } |m| \rrbracket P \\ & > [m \rightarrow M][s \rightarrow \{|M|\}_{j^-}] \text{case } s \text{ of } \{x\}_{k^+} \text{ in } \llbracket x \text{ is } |m| \rrbracket P \\ & > \text{case } \{|M|\}_{j^-} \text{ of } \{x\}_{k^+} \text{ in } [m \rightarrow M][s \rightarrow \{|M|\}_{k^-}] \llbracket x \text{ is } |m| \rrbracket P \end{aligned}$$

The process is stuck when $\{|M|\}_{j-}$ does not unify with $\{x\}_{k+}$.

Do we really want a signature check that fails to get stuck? I think so. M is available, but the signature check is stuck. A signed message is best represented as a pair $(M, \{|M|\}_{k-})$ allowing the message to be explicitly available.

$$\text{Struct Nil} \frac{}{P \mid \mathbf{0} \equiv P}$$

$$\text{Struct Comm} \frac{}{P \mid Q \equiv Q \mid P}$$

$$\text{Struct Assoc} \frac{}{P \mid (Q \mid R) \equiv (P \mid Q) \mid R}$$

$$\text{Struct Switch} \frac{}{(\nu m)(\nu n)P \equiv (\nu n)(\nu m)P}$$

$$\text{Struct Drop} \frac{}{(\nu n)\mathbf{0} \equiv \mathbf{0}}$$

$$\text{Struct Extrusion} \frac{n \notin fv(P)}{(\nu n)(P \mid Q) \equiv P \mid (\nu n)Q}$$

$$\text{Struct Red} \frac{P > Q}{P \equiv Q}$$

$$\text{Struct Refl} \frac{}{P \equiv P}$$

$$\text{Struct Symm} \frac{P \equiv Q}{Q \equiv P}$$

$$\text{Struct Trans} \frac{P \equiv Q \quad Q \equiv R}{P \equiv R}$$

$$\text{Struct Par} \frac{P \equiv P'}{P \mid Q \equiv P' \mid Q}$$

$$\text{Struct Res} \frac{P \equiv P'}{(\nu n)P \equiv (\nu n)P'}$$

$$\text{React Struct} \frac{P \equiv P' \quad P' \rightarrow Q' \quad Q' \equiv Q}{P \rightarrow Q}$$

$$\text{React Par} \frac{P' \rightarrow P'}{P \mid Q \rightarrow P' \mid Q}$$

$$\text{React Res} \frac{P' \rightarrow P'}{(\nu n)P \rightarrow (\nu n)P'}$$

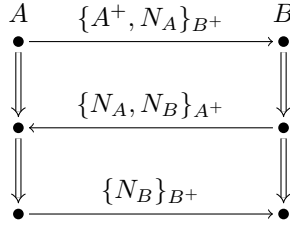
$$A \triangleq \bar{c}\langle\{(A, N.A)\}_{B^+}\rangle.$$

2.3 Privacy CA Protocol

3 Strand Spaces

3.1 Needham Schroder

$$\begin{aligned}
 A &\rightarrow B : \{A^+, N_A\}_{B^+} \text{ on } c \\
 B &\rightarrow A : \{N_A, N_B\}_{A^+} \text{ on } c \\
 A &\rightarrow B : \{N_B\}_{B^+} \text{ on } c
 \end{aligned}$$



A Glossary

- 0 - null process
- $|M|$ - hash of M
- K^+ - public half of asymmetric key K
- K^- - private half of asymmetric key K
- $\{M\}_K$ - encrypt M with symmetric key K
- $\{M\}_{K^+}$ - encrypt M with the public key from K
- $\{M\}_{K^-}$ - decrypt M with the public key from K
- $\{|M|\}_{K^-}$ - sign M with the private key from K
- $\{|M|\}_{K^+}$ - check signature on M with the public key from K
- $(\nu x)P$ - new variable x defined in scope of P
- $\bar{c}(M)$ - send M on channel c
- $c(M)$ - receive M on channel c
- $!P$ - infinite replication of P
- $P + Q$ - P or Q
- $P \mid Q$ - P in parallel with Q
- **case** $\{M\}_k$ of x in P - attempt to decrypt $\{M\}_k$ and bind to x in P if successful. Stuck if unsuccessful
- **case** $\{M\}_{k^-}$ of x in P - attempt to decrypt $\{M\}_{k^+}$ and bind to x in P if successful. Stuck if unsuccessful
- **case** $\{|M|\}_{k^+}$ of x in P - attempt to check signature $\{|M|\}_{k^-}$ and bind to x in P if successful. Stuck if unsuccessful

- $\text{case } x \text{ of } y \ 0 : P \ \text{succ}(x) : Q$ - case splitting over integers. x is bound in Q .
- $\text{let } (x, y) = M \text{ in } y$ - match M to (x, y) binding x and y to pair elements in M
- $A \triangleq B$ - define an equivalence
- $A \rightarrow B : M \text{ on } c$ - A sends B message M on channel c

$$A \triangleq (\nu c) \ \bar{c}\langle M \rangle. \mathbf{0} \mid c(M).A$$

References

- M. Abadi and A. Gordon. A Calculus for Cryptographic Protocols: The Spi Calculus. *Information and Computation*, 148:1–70, 1999.