Secrecy-preserving Refinement*

Jan Jürjens

Computing Laboratory, University of Oxford

www.jurjens.de/jan

*Work performed at Bell Labs Research, Lucent Technologies.

Security vs. Refinement

Goal: develop secure systems by stepwise refinement

from abstract specifications to concrete specifications.

Problem: common formulations of security properties are

not preserved by refinement ("refinement paradox") !

Applies in particular to implementations
(usually refinements of specifications) !!

Example (1)

$p \stackrel{\text{def}}{=} \mathit{if}\ H = 0\ \mathit{then}\ (\mathit{either}\ 0\ \mathit{or}\ 1)\ \mathit{else}\ (\mathit{either}\ 0\ \mathit{or}\ 1)$:

May seem to keep the value of H secret.

But: Not so for its refinement $p' \stackrel{\text{def}}{=} \mathit{if}\ H = 0\ \mathit{then}\ 0\ \mathit{else}\ 1$!

Example (2)

choose K_1 **or choose** K_2 **or** $\ldots$ **or choose** K_n

May seem secure (for large n).

But: Not so for its refinement **choose** K_1 !

This Work

- Use standard specification language with standard notions of refinement.
- Show that standard secrecy property is preserved by refinements.
- Apply to a proposed variant of TLS.

Specification language

Focus (M. Broy):

- Synchronously executing processes modelled by stream-processing functions.
- Interaction: transmission of data over unidirectional FIFO channels.
- Communication asynchronous (no refusal by receiver).

Extension: specification language with cryptographic primitives.

Specification Language: Expressions

Exp: empty expression ε and the non-empty expressions:

E ::=	expression
x	$x \in \mathbf{Var}$
c	$c \in \mathbf{Channels}$
d	$d \in \mathcal{D}$
K	key ($K \in \mathbf{Keys}$)
N	unguessable value ($N \in \mathbf{Secret}$)
$E_1 :: E_2$	concatenation
$\{E\}_e$	encryption ($e \in \mathbf{Enc}$)
$\mathcal{D}ec_e(E)$	decryption ($e \in \mathbf{Enc}$)

$\mathbf{Enc} \stackrel{\text{def}}{=} \mathbf{Keys} \cup \mathbf{Channels} \cup \mathbf{Var}$.

K^{-1}: decryption key corresponding to encryption key K.

Assume $\mathcal{D}ec_{K^{-1}}(\{E\}_K) = E$.

Programs

$p ::=$	program
$\quad E$	output expression
$\quad either\ p\ or\ q$	nondeterministic branching
$\quad if\ E = E'\ then\ p\ else\ q$	conditional
$\quad case\ E\ of\ key\ do\ p\ else\ q$	determine whether key
$\quad case\ E\ of\ x :: y\ do\ p\ else\ q$	break up list

Program computes one channel output at time $t + 1$
from input values at time t.

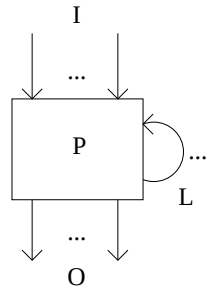
Iteration using local store.

Example $case\ c\ of\ key\ do\ \{d\}_c\ else\ \varepsilon$ outputs value from d
encrypted under value from c if it's a key, otherwise ε.

Processes

A process is of the form $P = (I, O, L, (p_c)_{c \in O \cup L})$ where

- $I \subseteq \mathbf{Channels}$ (input channels)
- $O \subseteq \mathbf{Channels}$ (output channels)
- $L \subseteq \mathbf{Channels}$ (local channels)
- p_c: closed program with input channels in $I \cup L$ and output channel $c \in O \cup L$.



Write $\tilde{I} \stackrel{\text{def}}{=} I \cup L$.

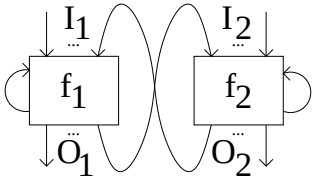
Stream-processing functions

$\mathbf{Stream}_C \stackrel{\text{def}}{=} (\mathbf{CExp}^\omega)^C$: C-indexed tuples of sequences of closed expressions.

For $f_i : \mathbf{Stream}_{I_i} \to \mathcal{P}(\mathbf{Stream}_{O_i})$ $(i = 1, 2)$ with $O_1 \cap O_2 = \emptyset$:

Define $f_1 \otimes f_2 : \mathbf{Stream}_I \to \mathcal{P}(\mathbf{Stream}_O)$

by $f_1 \otimes f_2(\vec{s}) = \{\vec{t}\restriction_O : \vec{t}\restriction_I = \vec{s}\restriction_I \wedge \vec{t}\restriction_{O_i} \in f_i(\vec{s}\restriction_{I_i})(i = 1, 2)\}$



(where $\vec{t} \in \mathbf{Stream}_{I \cup O}$, $I = (I_1 \cup I_2) \setminus (O_1 \cup O_2)$ and $O = (O_1 \cup O_2) \setminus (I_1 \cup I_2)$).

Model: Programs

Any closed p defines $[p] : \mathbf{CExp}^{\tilde{I}_p} \to \mathcal{P}(\mathbf{CExp})$:

- $[E](\vec{E}) = \{E(\vec{E})\}$
- $[either\ p\ or\ q](\vec{E}) = [p](\vec{E}) \cup [q](\vec{E})$
- $[if\ E = E'\ then\ p\ else\ q](\vec{E}) = [p](\vec{E})$ if $[E](\vec{E}) = [E'](\vec{E})$
- $[if\ E = E'\ then\ p\ else\ q](\vec{E}) = [q](\vec{E})$ if $[E](\vec{E}) \neq [E'](\vec{E})$
- $[case\ E\ of\ key\ do\ p\ else\ q](\vec{E}) = [p](\vec{E})$ if $[E](\vec{E}) \in \mathbf{Keys}$
- $[case\ E\ of\ key\ do\ p\ else\ q](\vec{E}) = [q](\vec{E})$ if $[E](\vec{E}) \notin \mathbf{Keys}$
- $[case\ E\ of\ x :: y\ do\ p\ else\ q](\vec{E}) = [p[h/x, t/y]](\vec{E})$ if $[E](\vec{E}) = h :: t$ with h atomic
- $[case\ E\ of\ x :: y\ do\ p\ else\ q](\vec{E}) = [q](\vec{E})$ if $[E](\vec{E}) = \varepsilon$

Model: Processes

Extend to streams per iteration:

Define $[p] : \mathbf{Stream}_{\tilde{I}} \to \mathcal{P}(\mathbf{Stream}_{\{c\}})$ by

$$[p](\vec{s}) \stackrel{\text{def}}{=} \{\vec{t} : \vec{t}_0 \in [p](\varepsilon, \ldots, \varepsilon) \wedge \forall n. \vec{t}_{n+1} \in [p](\vec{s}_n)\}.$$

Finally, $P = (I, O, L, (p_c)_{c \in \tilde{O}})$ gives $[\![P]\!] \stackrel{\text{def}}{=} \otimes_{c \in \tilde{O}} [p_c]$.

$[\![P]\!]$ causal: $n + 1^{st}$ output depends only on first n inputs.

Adversaries A may be non-causal: slightly different interpretation $[\![A]\!]_r$ (sometimes rushing adversaries, B. Pfitzmann).

Secrecy

f may eventually output E if
exists input-stream $\vec{s}$, output stream $\vec{t} \in f(\vec{s})$,
output channel c and point in time i such that $(\vec{t}(c))_i = E$.

Definition P preserves the secrecy of $m \in \mathbf{Secret} \cup \mathbf{Keys}$ if
exists no A such that $[\![P]\!] \otimes [\![A]\!]_r$ may eventually output m
(and $m \notin S_A \cup K_A$).

Protects atomic values (following Dolev, Yao 1983).

$p \stackrel{\text{def}}{=} \{m\}_K :: K$ does not preserve secrecy of m or K.
$p \stackrel{\text{def}}{=} \{m\}_K$ does.

Secrecy: Rely/guarantee

A obeys $C \subseteq \mathbf{Stream}_{O_P} \times \mathbf{Stream}_{I_P}$ if
for all $\vec{s} \in \mathbf{Stream}_{I_A}$ and $\vec{t} \in [\![A]\!](\vec{s})$, have $(\vec{s}\lfloor_O, \vec{t}\lfloor_I) \in C$.

Definition P preserves the secrecy of m assuming
$C \subseteq \mathbf{Stream}_{O_P} \times \mathbf{Stream}_{I_P}$ if exists no A obeying C such that
$[\![P]\!] \otimes [\![A]\!]_r$ may eventually output m (and $m \notin S_A \cup K_A$).

Example $p \stackrel{\text{def}}{=} \mathit{if}\ \ c = \mathbf{password}\ \mathit{then}\ \mathbf{secret}\ \mathit{else}\ \varepsilon$ preserves
secrecy of secret assuming $C = \{(\vec{t}, \vec{s}) : \forall n.\vec{s}_n \neq \mathbf{password}\}$.

Property Refinement

Definition Q refines P ($P \rightsquigarrow Q$) if for each $\vec{s} \in \mathbf{Stream}_{I_P}$ have $[\![P]\!](\vec{s}) \supseteq [\![Q]\!](\vec{s})$.

Example $(either\ p\ or\ q) \rightsquigarrow p$

Theorem

- If P preserves secrecy of m and $P \rightsquigarrow Q$ then Q preserves secrecy of m.
- If P preserves secrecy of m assuming C and $P \rightsquigarrow Q$ then Q preserves secrecy of m assuming C.

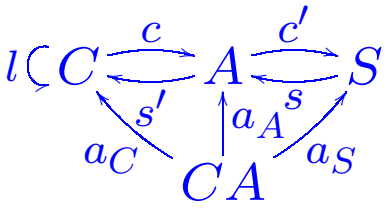
Why might this be true ?

Separate two kinds of non-determinism:

- underspecification ($either\ p\ or\ q$)
- unpredictability (key generation)

Proposed variant of the TLS (SSL) Handshake Protocol

Let client C send secret m to server S with confidentiality and server authentication.



$$C \xrightarrow{N_C::K_C::\mathcal{D}ec_{K_C^{-1}}(C::K_C)} S$$

$$C \xleftarrow{N_S::\{\mathcal{D}ec_{K_S^{-1}}(K_{CS}::N_C)\}_{K_C}::\mathcal{D}ec_{K_{CA}^{-1}}(S::K_S)} S$$

$$C \xrightarrow{\{m\}_{K_{CS}}} S$$

Apostolopoulos, Peris, Saha; IEEE Infocom 1999

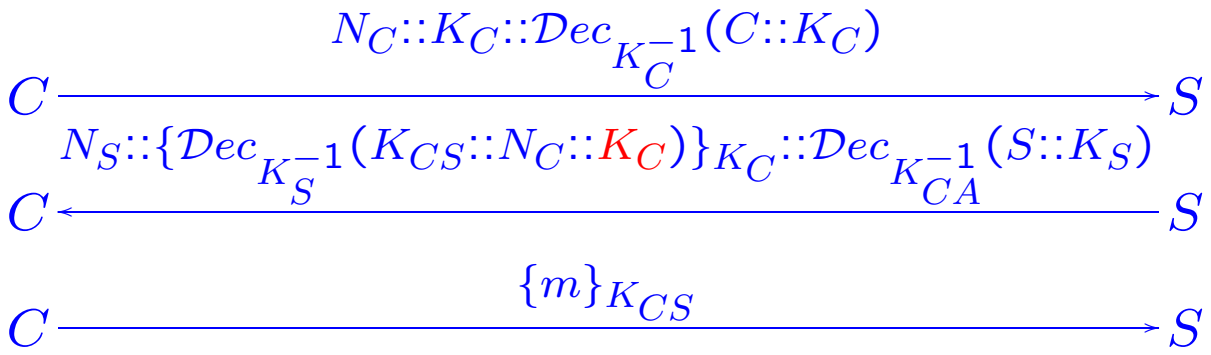
Specification

$$
\begin{aligned}
c \stackrel{\text{def}}{=}\ & \mathit{if}\ l = \varepsilon\ \mathit{then}\ N_C :: K_C :: \mathcal{D}ec_{K_C^{-1}}(C :: K_C) \\
& \quad \mathit{else}\ \mathit{case}\ s'\ \mathit{of}\ s_1 :: s_2 :: s_3 \\
& \quad\quad \mathit{do}\ \mathit{case}\ \mathcal{D}ec_{a_C}(s_3)\ \mathit{of}\ S :: x \\
& \quad\quad\quad \mathit{do}\ \mathit{if}\ \{\mathcal{D}ec_{K_C}(s_2)\}_x = y :: N_C\ \mathit{then}\ \{m\}_y\ \mathit{else}\ \mathsf{abort} \\
& \quad\quad\quad \mathit{else}\ \mathsf{abort} \\
& \quad\quad \mathit{else}\ \mathsf{abort} \\
l \stackrel{\text{def}}{=}\ & 0 \\
s \stackrel{\text{def}}{=}\ & \mathit{case}\ c'\ \mathit{of}\ c_1 :: c_2 :: c_3 \\
& \quad \mathit{do}\ \mathit{case}\ \{c_3\}_{c_2}\ \mathit{of}\ x :: c_2\ \mathit{do}\ N_S :: \{\mathcal{D}ec_{K_S^{-1}}(K_{CS} :: c_1)\}_{c_2} :: a_S\ \mathit{else}\ \varepsilon \\
& \quad \mathit{else}\ \mathsf{abort} \\
a_C \stackrel{\text{def}}{=}\ & K_{CA} \\
a_A \stackrel{\text{def}}{=}\ & K_{CA} \\
a_S \stackrel{\text{def}}{=}\ & \mathcal{D}ec_{K_{CA}^{-1}}(S :: K_S)
\end{aligned}
$$

Flaw and fix

Theorem $P \stackrel{\text{def}}{=} C \otimes S \otimes CA$ does not preserve secrecy of m.

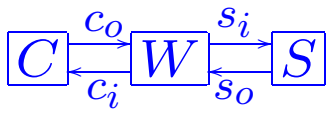
Man-in-the-middle attack. Correction:



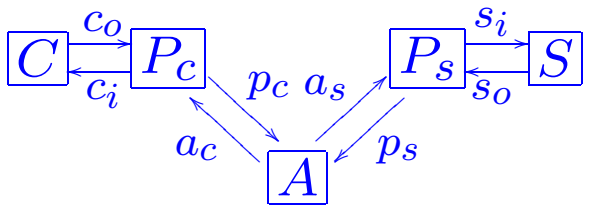
Theorem $P' \stackrel{\text{def}}{=} C' \otimes S' \otimes CA$ preserves secrecy of m.

Implementing Secure Channels

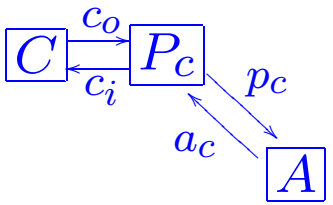
Want confidential channel W from C to S:



Transport layer vulnerable
against active attacks:



Implement using handshake protocol
(here client side).



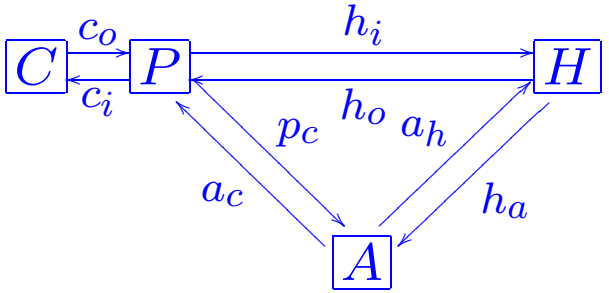
Want P_c so that for each suitable C: $C \otimes P_c$ sends out m
from C encrypted under negotiated key $K \in \mathbf{Keys}$
to network while preserving secrecy.

Channel (continued)

First: P_c nondeterministic sum of outputs needed for handshake and sending encrypted secret.

For any C: $C \otimes P_c$ preserves secrecy of m.

Next, split P_c into H (handshake protocol) and P (encrypts data from C under key from H and sends out to network):



Have conditional interface refinement from P_c to $P \otimes H$.

Use this to show that for each suitable C:
$C \otimes P \otimes H$ preserves secrecy of m.

Related Work

McLean (1996): Possibilistic security properties not in Alpern/Schneider framework.

S. Schneider (1996): Confidentiality property preserved under refinement. No cryptographic primitives considered.

Mantel (2001): Special refinement operators preserving secure information flow properties.

Conclusion

- Secrecy property preserved by various standard refinements.

- Used to uncover bug in TLS variant.

$\rightsquigarrow$ Theoretically sound notion of secrecy
with practical applicability.

Further Work

Other kinds of refinements from Focus (cf. proceedings)

Extension to other properties.

Compositionality (MMM'01)

Refinement of formal cryptographic primitives to complexity-theoretical treatment (joint with M. Abadi)

Foundation for developing secure systems with (formal core of) UML (FASE/ETAPS'01, WTUML'01, IFIP SEC'01,. . .)

Example (resolved)

$p \stackrel{\text{def}}{=} \mathit{if}\ H = 0\ \mathit{then}\ (\mathit{either}\ 0\ \mathit{or}\ 1)\ \mathit{else}\ (\mathit{either}\ 0\ \mathit{or}\ 1)$:

Does not keep the value of H secret.

Interpret nondeterminism as underspecification, not as possibilistic nondeterminism.

Interface refinement

Definition 1 P_1, P_2, D, U processes with $I_{P_1} = I_D$, $O_D = I_{P_2}$, $O_{P_2} = I_U$ and $O_U = O_{P_1}$. Define $P_1 \overset{(D,U)}{\rightsquigarrow} P_2$ if $P_1 \rightsquigarrow D \otimes P_2 \otimes U$.

Theorem 1 *P_1, P_2, D, U processes with $I_{P_1} = I_D$, $O_D = I_{P_2}$, $O_{P_2} = I_U$ and $O_U = O_{P_1}$ such that D has right inverse D' and U left inverse U'.*

- *If $P_1(m)$ preserves secrecy of m and $P_1 \overset{(D,U)}{\rightsquigarrow} P_2$ then $P_2(m)$ preserves secrecy of m.*

- *If $P_1(m)$ preserves secrecy of m assuming $C \subseteq \mathbf{Stream}_{O_{P_1}} \times \mathbf{Stream}_{I_{P_1}}$ and $P_1 \overset{(D,U)}{\rightsquigarrow} P_2$ then $P_2(m)$ preserves secrecy of m assuming $U' \circ C \circ D'$.*

Notions of Processes

For $\vec{s} \in \mathbf{Stream}_X$ and bijection $\iota : Y \to X$ define $\vec{s}_\iota \in \mathbf{Stream}_Y$ by $\vec{s}_\iota(y) = \vec{s}(\iota(y))$.

For processes D, D' with $O_D = I_{D'}$ and $O_{D'} \cap I_D = \emptyset$ and bijection $\iota : O_{D'} \to I_D$ with $[\![D]\!] \otimes [\![D']\!](\vec{s}) = \{\vec{s}_\iota\}$ for each $\vec{s} \in \mathbf{Stream}_{I_D}$, D is a right inverse of D' and D' is a left inverse of D.

Define $S \circ R = \{(x, z) : \exists y.(x, y) \in R \wedge (y, z) \in S\}$ (usual composition of relations R, S).

For function $F : X \to \mathcal{P}(Y)$ and relation $C \subseteq Y \times X$ define $F\wr_C \stackrel{\text{def}}{=} F \cap C^{-1}$ where $C^{-1} = \{(x, y) : (y, x) \in C\}$.

Conditional refinement

Definition 2 For processes P_1, P_2 with $I_{P_1} = I_{P_2}$ and $O_{P_1} = O_{P_2}$ define $P_1 \rightsquigarrow_C P_2$ (for total relation $C \subseteq \mathbf{Stream}_{O_{P_1}} \times \mathbf{Stream}_{I_{P_1}}$) if $[\![P_1]\!]\wr_C \supseteq [\![P_2]\!]\wr_C$.

Theorem 2

For total relations $C, D \subseteq \mathbf{Stream}_{O_P} \times \mathbf{Stream}_{I_P}$ with $C \subseteq D$, if $P(m)$ preserves secrecy of m assuming C and $P \rightsquigarrow_D Q$ then $Q(m)$ preserves secrecy of m assuming C.

The flaw

Theorem 3 $P \stackrel{\text{def}}{=} C \otimes S \otimes CA$ *does not preserve secrecy of* m.

Attacker A:

$$c' \stackrel{\text{def}}{=} \mathit{case}\ c\ \mathit{of}\ c_1 :: c_2 :: c_3\ \mathit{do}\ c_1 :: K_A :: \mathcal{D}ec_{K_A^{-1}}(C :: K_A)\ \mathit{else}\ \varepsilon$$

$$s' \stackrel{\text{def}}{=} \mathit{case}\ s\ \mathit{of}\ s_1 :: s_2 :: s_3\ \mathit{do}\ s_1 :: \{\mathcal{D}ec_{K_A^{-1}}(s_2)\}_{K_C} :: s_3\ \mathit{else}\ \varepsilon$$

$$l_A \stackrel{\text{def}}{=} \mathit{if}\ l_A = \varepsilon\ \mathit{then}\ \mathit{case}\ s\ \mathit{of}\ s_1 :: s_2 :: s_3\ \mathit{do}\ \mathit{case}\ \{\mathcal{D}ec_{K_A^{-1}}(s_2)\}_{K_S}\ \mathit{of}\ x_1 :: x_2\ \mathit{do}\ x_1\ \mathit{else}\ l_A\ \mathit{else}\ l_A$$

$$c_0 \stackrel{\text{def}}{=} \mathit{case}\ l_A\ \mathit{of}\ \mathit{key}\ \mathit{do}\ \mathit{if}\ \mathcal{D}ec_{l_A}(c) = \perp\ \mathit{then}\ \varepsilon\ \mathit{else}\ \mathcal{D}ec_{l_A}(c)\ \mathit{else}\ \varepsilon$$

Channel: Secrecy

First step: P_c nondeterministic sum of possible outputs.

$$p_c \stackrel{\text{def}}{=} \mathit{either}\ \mathit{if}\ c_o = \varepsilon\ \mathit{then}\ \varepsilon\ \mathit{else}\ \{c_o\}_K \ \mathit{or}\ c_K$$

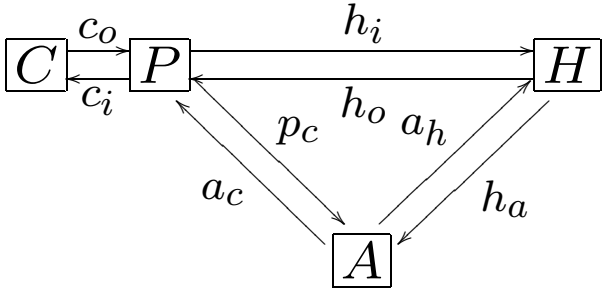
$$c_i \stackrel{\text{def}}{=} \mathit{either}\ \varepsilon\ \mathit{or}\ \mathsf{ok}$$

$$\begin{aligned} c_K \stackrel{\text{def}}{=}\ & \mathit{either}\ N_C :: K_C :: \mathcal{D}ec_{K_C^{-1}}(C :: K_C) \\ & \mathit{or}\ \mathit{case}\ a_c\ \mathit{of}\ s_1 :: s_2 :: s_3 \\ & \quad \mathit{do}\ \mathit{case}\ \mathcal{D}ec_{K_{AC}}(s_3)\ \mathit{of}\ S :: x \\ & \quad\quad \mathit{do}\ \mathit{if}\ \{\mathcal{D}ec_{K_C}(s_2)\}_x = y :: N_C :: K_C\ \mathit{then}\ \{K\}_y\ \mathit{else}\ \mathsf{abort} \\ & \quad\quad \mathit{else}\ \mathsf{abort} \\ & \quad \mathit{else}\ \mathsf{abort} \end{aligned}$$

Channel: Concrete Spec

For any $C(n)$, $C(n) \otimes P_c$ preserves secrecy of n.

Next, split P_c into H and P:



$$h_a \stackrel{\text{def}}{=} \textit{if } h_i = \varepsilon \textit{ then } N_C :: K_C :: \mathcal{D}ec_{K_C^{-1}}(C :: K_C)$$
$$\textit{else case } a_h \textit{ of } s_1 :: s_2 :: s_3$$
$$\textit{do case } \mathcal{D}ec_{K_{AC}}(s_3) \textit{ of } S :: x$$
$$\textit{do if } \{\mathcal{D}ec_{K_C}(s_2)\}_x = y :: N_C :: K_C \textit{ then } \{m\}_y \textit{ else abort}$$
$$\textit{else abort}$$
$$\textit{else abort}$$

$$
\begin{array}{rcl}
h_o & \stackrel{\text{def}}{=} & \mathit{if}\ h_i = \varepsilon\ \mathit{then}\ \varepsilon \\
& & \quad \mathit{else}\ \mathit{case}\ a_h\ \mathit{of}\ s_1 :: s_2 :: s_3 \\
& & \quad\quad \mathit{do}\ \mathit{case}\ \mathcal{D}ec_{K_{AC}}(s_3)\ \mathit{of}\ S :: x \\
& & \quad\quad\quad \mathit{do}\ \mathit{if}\ \{\mathcal{D}ec_{K_C}(s_2)\}_x = y :: N_C :: K_C\ \mathit{then}\ \mathsf{finished}\ \mathit{else}\ \varepsilon \\
& & \quad\quad\quad \mathit{else}\ \varepsilon \\
& & \quad\quad \mathit{else}\ \varepsilon \\
h_i & \stackrel{\text{def}}{=} & 0 \\
p_c & \stackrel{\text{def}}{=} & \mathit{if}\ c_o = \varepsilon\ \mathit{then}\ \varepsilon\ \mathit{else}\ \{c_o\}_K \\
c_i & \stackrel{\text{def}}{=} & \mathit{if}\ h_o = \mathsf{finished}\ \mathit{then}\ \mathsf{ok}\ \mathit{else}\ \varepsilon
\end{array}
$$

Channel: Refinement

Have conditional interface refinement $P_c \overset{(D,U)}{\rightsquigarrow}_T P \otimes H$ where

$T = \{(\vec{s}, \vec{t}) : \forall n.(\forall i \leq n(\vec{s}(\tilde{c}_i)) \downarrow_i \neq finished \Rightarrow \forall i \leq n+1.(\vec{s}(\tilde{c}_o)) \downarrow_i = \varepsilon)\}$

$((\vec{s}, \vec{t}) \in \mathbf{Stream}_{O_{P_c}} \times \mathbf{Stream}_{I_{P_c}})$ and
U, D given by $I_D = \{\tilde{c}_o, \tilde{a}_c\}$, $O_D = \{c_o, a_c, a_h\}$, $I_U = \{c_i, p_c, h_a\}$ and $O_U = \{\tilde{c}_i, \tilde{p}_c\}$ and $c_o \overset{\text{def}}{=} \tilde{c}_o, a_c \overset{\text{def}}{=} \tilde{a}_c, a_h \overset{\text{def}}{=} \tilde{a}_c, \tilde{c}_i \overset{\text{def}}{=} c_i, \tilde{p}_c \overset{\text{def}}{=} h_a$ (after renaming channels of P_c to $\tilde{c}_o, \tilde{c}_i, \tilde{p}_c, \tilde{a}_c$).

Thus for $C(n)$ with $[\![C(n)]\!] \subseteq T$, have interface refinement $C(n) \otimes P_c \overset{(D,U)}{\rightsquigarrow} C(n) \otimes P \otimes H$. So for any $C(n)$ with $[\![C(n)]\!] \subseteq T$, $C(n) \otimes P \otimes H$ preserves secrecy of n.