FMSP: Solution Attempt for exam 2020-09

Pizzaiolo

June 28, 2022

Preamble

Feel free to edit it to add more examples, fix mistakes, etc. If you do so, please update both the .tex and .pdf files. And obviously no guarantee for anything.

Problem 1

a)

Does following program perserve secrecy and/or integrity? If yes, give a type derivation. If not, give a counter-example.

$$(\nu k_1 \colon DecK^{HH}[LL, HL]).c(x).\textbf{case } x \textbf{ of } \{|[y,z]|\}^a_{k_1} \textbf{ in } (\nu k_2 \colon DecK^{HH}[LL, HL, HH]).(\nu v_1 \colon HH).\bar{c}\langle\{|[y,z,v_1]|\}^a_{ek(k_2)}\rangle.0$$

Type derivation We typecheck for $\Gamma = \{x \colon LL, y \colon LL, z \colon LL\, c \colon LL\}$.
For simplicity let

$$P = (\nu k_2 \colon DecK^{HH}[LL, HL, HH]).(\nu v_1 \colon HH).\bar{c}\langle\{|[y,z,v_1]|\}^a_{ek(k_2)}\rangle.0$$

$$\Gamma_1 = \Gamma, k_1 \colon DecK^{HH}[LL, HL], x \colon LL$$

.

$$\dfrac{\dfrac{\dfrac{(1)}{\Gamma_1 \vdash \textbf{case } x \textbf{ of } \{|[y,z]|\}^a_{k_1} \textbf{ in } P} \quad \Gamma, k_1 \colon DecK^{HH}[LL, HL] \vdash c \colon C^{LL}[LL]}{\Gamma, k_1 \colon DecK^{HH}[LL, HL] \vdash c(x).\textbf{case } x \textbf{ of } \{|[y,z]|\}^a_{k_1} \textbf{ in } P}\ \text{IN}}{\Gamma \vdash (\nu k_1 \colon DecK^{HH}[LL, HL]).c(x).\textbf{case } x \textbf{ of } \{|[y,z]|\}^a_{k_1} \textbf{ in } P}\ \text{RES}$$

Decryption .

$$\dfrac{\dfrac{\dfrac{(2)}{\Gamma_1 \vdash x \colon LL} \quad \dfrac{(3)}{\Gamma_1 \vdash k_1 \colon DecK^{HH}[LL, HL]} \quad \dfrac{(4)}{\Gamma_1, [y,z] \colon [LL, HL] \vdash P} \quad \dfrac{(5)}{\mathcal{L}(LL) = L \implies \Gamma_1, [y,z] \colon LL \vdash P}\ \text{AsymDec}}{\Gamma_1 \vdash \textbf{case } x \textbf{ of } \{|[y,z]|\}^a_{k_1} \textbf{ in } P}}{(1)}$$

Typecheck x and key type $\dfrac{\dfrac{(0.1)}{\Gamma_1 \vdash \diamond} \quad x\colon LL \text{ in } \Gamma_1}{\dfrac{\Gamma_1 \vdash x\colon LL}{(2)}}$ ATOM

$$\dfrac{\dfrac{(0.1)}{\Gamma_1 \vdash \diamond} \quad k_1\colon DecK^{HH}[LL, HL] \text{ in } \Gamma_1}{\dfrac{\Gamma_1 \vdash k_1\colon DecK^{HH}[LL, HL]}{(3)}}\ \text{ATOM}$$

Typechecking (4) .

$$\dfrac{\dfrac{\dfrac{\dfrac{(6)}{\Gamma_1, [y,z]\colon [LL,HL], (k_2\colon DecK^{HH}[LL,HL,HH]), (v_1\colon HH) \vdash \overline{c}\langle\{|[y,z,v_1]|\}^a_{ek(k_2)}\rangle.0}\ \text{RES}}{\Gamma_1, [y,z]\colon [LL,HL], (k_2\colon DecK^{HH}[LL,HL,HH]) \vdash (\nu v_1\colon HH).\overline{c}\langle\{|[y,z,v_1]|\}^a_{ek(k_2)}\rangle.0}\ \text{RES}}{\Gamma_1, [y,z]\colon [LL,HL] \vdash (\nu k_2\colon DecK^{HH}[LL,HL,HH]).(\nu v_1\colon HH).\overline{c}\langle\{|[y,z,v_1]|\}^a_{ek(k_2)}\rangle.0}}{\dfrac{\Gamma_1, [y,z]\colon [LL,HL] \vdash P}{(4)}}$$

Now let $\Gamma_2 = \Gamma_1, [y,z]\colon [LL,HL], (k_2\colon DecK^{HH}[LL,HL,HH]), (v_1\colon HH)$

$$\dfrac{\dfrac{\dfrac{(7)}{\Gamma_2 \vdash \{|[y,z,v_1]|\}^a_{ek(k_2)}\colon LL} \quad \dfrac{\dfrac{(0.2)}{\Gamma_2 \vdash \diamond}}{\Gamma_2 \vdash 0}\ \text{STOP} \qquad \dfrac{\dfrac{\dfrac{(0.2)}{\Gamma_2 \vdash \diamond} \quad c\colon LL \text{ in } \Gamma_2}{\Gamma_2 \vdash c\colon LL}\ \text{ATOM} \qquad LL \leq C^{LL}[LL]}{\Gamma_2 \vdash c\colon C^{LL}[LL]}\ \text{SUBSUMPTION}}{\Gamma_2 \vdash \overline{c}\langle\{|[y,z,v_1]|\}^a_{ek(k_2)}\rangle.0}\ \text{OUT}}{(6)}$$

Typechecking the ciphertext .

$$\dfrac{\dfrac{\dfrac{(0.2)}{\Gamma_2 \vdash \diamond} \quad k_2\colon EncKey^{HH}[LL,HL,HH] \text{ in } \Gamma_2}{\Gamma_2 \vdash k_2\colon EncKey^{HH}[LL,HL,HH]}\ \text{ATOM} \qquad \dfrac{(8)}{\Gamma_2 \vdash [y,z,v_1]\colon [LL,HL,HH]}}{\Gamma_2 \vdash \{|[y,z,v_1]|\}^a_{ek(k_2)}\colon LH}\ \text{AsymEnc} \qquad LH \leq LL}{\dfrac{\Gamma_2 \vdash \{|[y,z,v_1]|\}^a_{ek(k_2)}\colon LL}{(7)}}\ \text{SUBSUMPTION}$$

Message type .

$$\dfrac{\dfrac{\dfrac{(0.2)}{\Gamma_2 \vdash \diamond} \quad y\colon LL \text{ in } \Gamma_2}{\Gamma_2 \vdash y\colon LL}\ \text{ATOM} \quad \dfrac{\dfrac{(0.2)}{\Gamma_2 \vdash \diamond} \quad z\colon HL \text{ in } \Gamma_2}{\Gamma_2 \vdash z\colon HL}\ \text{ATOM} \quad \dfrac{\dfrac{(0.2)}{\Gamma_2 \vdash \diamond} \quad v_1\colon HH \text{ in } \Gamma_2}{\Gamma_2 \vdash v_1\colon HH}\ \text{ATOM}}{\Gamma_2 \vdash [y, z, v_1]\colon [LL, HL, HH]}\ \text{LIST}}{(8)}$$

Proving (5) It holds that $\mathcal{L}(LL) = L$, therefore we need to prove $\Gamma_1, [y, z]\colon LL \vdash P$. This is analog to (4), except that we have $z\colon LL$ instead of $z\colon HL$. Therefore I skip everything until this difference matteres. Accordingly let

$$\Gamma_3 = \Gamma_1, [y, z]\colon [LL, LL], (k_2\colon DecK^{HH}[LL, HL, HH]), (v_1\colon HH)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(0.3)}{\Gamma_3 \vdash \diamond} \quad z\colon LL \text{ in } \Gamma_3}{\Gamma_3 \vdash z\colon LL}\ \text{ATOM} \quad LL \leq HL}{\Gamma_3 \vdash z\colon HL}\ \text{SUBSUMPTION}}{\text{analog to (4)}}}{\Gamma_1, [y, z]\colon LL \vdash P}}{\mathcal{L}(LL) = L \implies \Gamma_1, [y, z]\colon LL \vdash P}}{(5)}$$

Proving wellformedness Not proving it now cause too lazy. Essentially, we have no channels and all keys are of type k^{HH}, thus we can use the ENV rule.

b)

c)

Probably

$$(\nu m\colon HL).(\nu c\colon C^{HH}[HH]).\bar{c}\langle m\rangle$$

Problem 3: Information Flow

(a)

We have the program

$$\text{while } l > 0 \text{ do } \{ \text{ if } h = 0 \text{ then } \{h := h + 1\} \text{ else } \{ \text{ skip } \}; l := l - 1\}; h := 1$$

Typechecking .

$$\dfrac{(1) \quad \dfrac{todo}{\Gamma \vdash h := l}\ \text{ASSIGN}}{\Gamma \vdash \text{ while } l > 0 \text{ do } \{ \text{ if } h = 0 \text{ then } \{h := h + 1\} \text{ else } \{ \text{ skip } \}; l := l - 1\}; h := l\colon L \text{ cmd}}\ \text{COMPOSE}$$

$$\dfrac{\dfrac{(2)}{\Gamma \vdash l > 0 \colon L} \quad \dfrac{\dfrac{(3)}{\Gamma \vdash \text{ if } h = 0 \text{ then } \{h \coloneqq h + 1\} \text{ else } \{ \text{ skip } \} \colon L \text{ cmd}} \quad \dfrac{(4)}{\Gamma \vdash l \coloneqq l - 1 \colon L \text{ cmd}}\ \text{COMPOSE}}{\Gamma \vdash \text{ if } h = 0 \text{ then } \{h \coloneqq h + 1\} \text{ else } \{ \text{ skip } \}; l \coloneqq l - 1 \colon L \text{ cmd}}\ \text{WHILE}}{\Gamma \vdash \text{ while } l > 0 \text{ do } \{ \text{ if } h = 0 \text{ then } \{h \coloneqq h + 1\} \text{ else } \{ \text{ skip } \}; l \coloneqq l - 1\} \colon L \text{ cmd}}$$
(1)

Condition $l > 0$

$$\dfrac{\dfrac{\dfrac{\Gamma(l) = L \text{ var}}{\Gamma \vdash l \colon L}\ \text{R-Val} \quad \dfrac{}{\Gamma \vdash 0 \colon L}\ \text{INT} \qquad > \in \{+, -, >, <\}}{\Gamma \vdash l > 0 \colon L}\ \text{OP}}{(2)}$$

Assignment $l \coloneqq l - 1$

$$\dfrac{\Gamma(l) = L \text{ var} \qquad \dfrac{\dfrac{\dfrac{\Gamma(l) = L \text{ var}}{\Gamma \vdash l \colon L}\ \text{R-Val} \quad \dfrac{}{\Gamma \vdash 1 \colon L}\ \text{INT} \qquad - \in \{+, -, >, <\}}{\Gamma \vdash l - 1 \colon L}\ \text{OP}}{\Gamma \vdash l \coloneqq l - 1 \colon L \text{ cmd}}\ \text{ASSIGN}}{(4)}$$

If condition

$$\dfrac{\dfrac{\dfrac{(5)}{\Gamma \vdash h = 0 \colon H} \quad \dfrac{(6)}{\Gamma \vdash h \coloneqq h + 1 \colon H \text{ cmd}} \quad \dfrac{}{\Gamma \vdash \text{ skip } \colon H \text{ cmd}}\ \text{SKIP}}{\Gamma \vdash \text{ if } h = 0 \text{ then } \{h \coloneqq h + 1\} \text{ else } \{ \text{ skip } \} \colon H \text{ cmd}}\ \text{IF} \qquad \dfrac{}{H \text{ cmd} \subseteq L \text{ cmd}}\ \text{CMD- and BASE SUBSUMP}}{\Gamma \vdash \text{ if } h = 0 \text{ then } \{h \coloneqq h + 1\} \text{ else } \{ \text{ skip } \} \colon L \text{ cmd}}$$
(3)

Condition type

$$\dfrac{\dfrac{\text{either fails here or missing operator} =}{\Gamma \vdash h = 0 \colon H}}{(5)}$$

$$\dfrac{\Gamma(h) = H \text{ var} \qquad \dfrac{\dfrac{\dfrac{\Gamma(h) = H \text{ var}}{\Gamma \vdash h \colon H}\ \text{R-Val} \quad \dfrac{}{\Gamma \vdash 1 \colon H}\ \text{SUBSUMPTION + INT} \qquad + \in \{+, -, >, <\}}{\Gamma \vdash h + 1 \colon H}\ \text{OP}}{\Gamma \vdash h \coloneqq h + 1 \colon H \text{ cmd}}\ \text{ASSIGN}}{(6)}$$

(b)

No, the program while $h = 1$ do skip does typecheck, but it is not termination sensitive non-interferent (if it terminates, the attacker knows that $h \neq 1$)