A formal Analysis for Capturing Replay Attacks in Cryptographic Protocols

Han Gao, Chiara Bodei, Pierpaolo Degano, y Hanne Riis Nielson

Katherine Sullivan
FCEIA - UNR

Índice

Índice

Introducción

Introducción

- ¿En qué consisten los ataques por repetición?

Introducción

- ¿En qué consisten los ataques por repetición?
  - Tipo de ataque en el que un adversario intercepta y retransmite datos previamente capturados para intentar ganar acceso no autorizado o causar un mal funcionamiento en un sistema.

Introducción

- ¿En qué consisten los ataques por repetición?
  - Tipo de ataque en el que un adversario intercepta y retransmite datos previamente capturados para intentar ganar acceso no autorizado o causar un mal funcionamiento en un sistema.

- ¿Cómo se hará el análisis formal para capturar ataques por repetición?

Introducción

- ¿En qué consisten los ataques por repetición?
  - Tipo de ataque en el que un adversario intercepta y retransmite datos previamente capturados para intentar ganar acceso no autorizado o causar un mal funcionamiento en un sistema.

- ¿Cómo se hará el análisis formal para capturar ataques por repetición?
  - A través de la extensión de LySA, un álgebra de procesos, y su respectivo análisis de flujo de control, con anotaciones de sesiones.

Índice

¿Qué es LySA?

¿Qué es LySA?

Es un álgebra de procesos desarrollada en Automatic Validation of Protocol Narration (2003) y en Static Validation of Security Protocols (2005) por Chiara Bodei, Mikael Buchholtz, Pierpaolo Degano, Flemming Nielson y Hanne Riis Nielson con ciertas particularidades:

¿Qué es LySA?

Es un álgebra de procesos desarrollada en Automatic Validation of Protocol Narration (2003) y en Static Validation of Security Protocols (2005) por Chiara Bodei, Mikael Buchholtz, Pierpaolo Degano, Flemming Nielson y Hanne Riis Nielson con ciertas particularidades:

- No hay canales: en LySA todos los procesos tienen acceso solo a un único canal de comunicación global.

¿Qué es LySA?

Es un álgebra de procesos desarrollada en Automatic Validation of Protocol Narration (2003) y en Static Validation of Security Protocols (2005) por Chiara Bodei, Mikael Buchholtz, Pierpaolo Degano, Flemming Nielson y Hanne Riis Nielson con ciertas particularidades:

- No hay canales: en LySA todos los procesos tienen acceso solo a un único canal de comunicación global.
- Las verificaciones asociadas con *inputs* (recepciones de mensajes) y desencriptaciones son expresadas usando *pattern matching*.

Sintaxis LySA

Sintaxis LySA

La sintaxis de expresiones resulta simple de comprender, estando conformada por nombres, variables y expresiones encriptadas. Vale la pena detenerse en la sintaxis de procesos.

Sintaxis LySA

La sintaxis de expresiones resulta simple de comprender, estando conformada por nombres, variables y expresiones encriptadas. Vale la pena detenerse en la sintaxis de procesos.

$$E ::= n | x | \{E_1, ..., E_k\}_{E_0}$$

$$\begin{aligned}
P ::= & \ \langle E_1, \ldots, E_k \rangle.P && \text{(envío de msj)} \\
| & \ (E_1, \ldots, E_j; x_{j+1}, \ldots, x_k).P && \text{(recepción de msj)} \\
| & \ \text{decrypt } E \text{ as } \{E_1, \ldots, E_j; x_{j+1}, \ldots, x_k\}^{I}_{E_0} \text{ in } P && \text{(desencriptación)} \\
| & \ (\nu n)P && \text{(nuevo nombre)} \\
| & \ P_1 \mid P_2 && \text{(paralelismo)} \\
| & \ !P && \text{(replicación)} \\
| & \ 0 && \text{(proceso nulo)}
\end{aligned}$$

Sintaxis LySA extendida I

Sintaxis LySA extendida I

Ahora cada término y proceso llevará un identificador de la sesión a la que pertenece.

Sintaxis LySA extendida I

Ahora cada término y proceso llevará un identificador de la sesión a la que pertenece.

$$
\begin{aligned}
\mathcal{E} &::= [n]_s \mid x \mid [\{\mathcal{E}_1, \ldots, \mathcal{E}_k\}_{\mathcal{E}_0}]_s \\
\mathcal{P} &::= \langle \mathcal{E}_1, \ldots, \mathcal{E}_k \rangle.\mathcal{P} \mid (\mathcal{E}_1, \ldots, \mathcal{E}_j; x_{j+1}, \ldots, x_k).\mathcal{P} \mid \\
&\quad \text{decrypt } \mathcal{E} \text{ as } \{\mathcal{E}_1, \ldots, \mathcal{E}_j; x_{j+1}, \ldots, x_k\}^l_{\mathcal{E}_0} \text{ in } \mathcal{P} \mid \\
&\quad (\nu\ [n]_s)\mathcal{P} \mid \mathcal{P}_1 | \mathcal{P}_2 \mid [!P]_s \mid 0
\end{aligned}
$$

Sintaxis LySA extendida I

Ahora cada término y proceso llevará un identificador de la sesión a la que pertenece.

$$
\begin{aligned}
\mathcal{E} &::= [n]_s \mid x \mid [\{\mathcal{E}_1, \ldots, \mathcal{E}_k\}_{\mathcal{E}_0}]_s \\
\mathcal{P} &::= \langle \mathcal{E}_1, \ldots, \mathcal{E}_k \rangle.\mathcal{P} \mid (\mathcal{E}_1, \ldots, \mathcal{E}_j; x_{j+1}, \ldots, x_k).\mathcal{P} \mid \\
&\quad \text{decrypt } \mathcal{E} \text{ as } \{\mathcal{E}_1, \ldots, \mathcal{E}_j; x_{j+1}, \ldots, x_k\}^l_{\mathcal{E}_0} \text{ in } \mathcal{P} \mid \\
&\quad (\nu\ [n]_s)\mathcal{P} \mid \mathcal{P}_1 | \mathcal{P}_2 \mid [!P]_s \mid 0
\end{aligned}
$$

Pero, ¿cómo se mapean términos y procesos estándar a unos de la sintaxis extendida?

Sintaxis LySA extendida I

Ahora cada término y proceso llevará un identificador de la sesión a la que pertenece.

$$
\begin{aligned}
\mathcal{E} &::= [n]_s \mid x \mid [\{\mathcal{E}_1, \ldots, \mathcal{E}_k\}_{\mathcal{E}_0}]_s \\
\mathcal{P} &::= \langle \mathcal{E}_1, \ldots, \mathcal{E}_k \rangle.\mathcal{P} \mid (\mathcal{E}_1, \ldots, \mathcal{E}_j; x_{j+1}, \ldots, x_k).\mathcal{P} \mid \\
&\quad \text{decrypt } \mathcal{E} \text{ as } \{\mathcal{E}_1, \ldots, \mathcal{E}_j; x_{j+1}, \ldots, x_k\}^l_{\mathcal{E}_0} \text{ in } \mathcal{P} \mid \\
&\quad (\nu\ [n]_s)\mathcal{P} \mid \mathcal{P}_1 | \mathcal{P}_2 \mid [!P]_s \mid 0
\end{aligned}
$$

Pero, ¿cómo se mapean términos y procesos estándar a unos de la sintaxis extendida? Añadiendo identificadores de sesión inductivamente a través de dos funciones: $\mathcal{F}$ y $\mathcal{T}$.

Sintaxis LySA extendida II

Sintaxis LySA extendida II

$$\mathcal{F} : E \times SID \to \mathcal{E}$$

$$-\mathcal{F}(n, s) = [n]_s \qquad -\mathcal{F}(x, s) = x$$
$$-\mathcal{F}(\{E_1, \ldots, E_k\}_{E_0}, s) = [\{\mathcal{F}(E_1, s), \ldots, \mathcal{F}(E_k, s)\}_{\mathcal{F}(E_0,s)}]_s$$

$$\mathcal{T} : P \times SID \to \mathcal{P}$$

$$-\mathcal{T}(\langle E_1, \ldots, E_k\rangle.P, s) = \langle \mathcal{F}(E_1, s), \ldots, \mathcal{F}(E_k, s)\rangle.\mathcal{T}(P, s)$$
$$-\mathcal{T}((E_1, \ldots, E_j; x_{j+1}, \ldots, x_k).P, s) =$$
$$(\mathcal{F}(E_1, s), \ldots, \mathcal{F}(E_j, s); x_{j+1}, \ldots, x_k).\mathcal{T}(P, s)$$

$$-\mathcal{T}(\textit{decrypt } E \textit{ as } \{E_1, \ldots, E_j; x_{j+1}, \ldots, x_k\}^l_{E_0} \textit{ in } P, s) =$$
$$\textit{decrypt } \mathcal{F}(E, s) \textit{ as } \{\mathcal{F}(E_1, s), \ldots, \mathcal{F}(E_j, s); x_{j+1}, \ldots, x_k\}^l_{\mathcal{F}(E_0,s)} \textit{ in } \mathcal{T}(P, s)$$

$$-\mathcal{T}(P \mid Q, s) = \mathcal{T}(P, s) \mid \mathcal{T}(Q, s) \qquad -\mathcal{T}((\nu\ n)P, s) = (\nu\ [n]_s)\mathcal{T}(P, s)$$
$$-\mathcal{T}(!P, s) = [!P]_s \qquad -\mathcal{T}(0, s) = 0$$

Semántica operacional I

Semántica operacional I

Se consideran dos variantes de la relación de reducción $\rightarrow_{\mathcal{R}}$, identificadas por una diferente instanciación de la relación R, que decora la relación de transición.

Semántica operacional I

Se consideran dos variantes de la relación de reducción $\rightarrow_{\mathcal{R}}$, identificadas por una diferente instanciación de la relación R, que decora la relación de transición.

Una variante ($\rightarrow_{RM}$) aprovecha las anotaciones, la otra ($\rightarrow$) las descarta: esencialmente, la primera semántica verifica la frescura de los mensajes, mientras que la otra no lo hace.

Semántica operacional II

Semántica operacional II

Antes de pasar a la definición de la relación necesitamos de dos definiciones:

Semántica operacional II

Antes de pasar a la definición de la relación necesitamos de dos definiciones:

Semántica operacional II

Antes de pasar a la definición de la relación necesitamos de dos definiciones:

- La relación de equivalencia $V_1 \stackrel{f}{=} V_2$ definida como la menor equivalencia sobre *Val* que (de manera inductiva) ignora los identificadores de sesión.

Semántica operacional II

Antes de pasar a la definición de la relación necesitamos de dos definiciones:

- La relación de equivalencia $V_1 \stackrel{f}{=} V_2$ definida como la menor equivalencia sobre Val que (de manera inductiva) ignora los identificadores de sesión.
- La función $\mathcal{I} : Val \to SID$ de extracción de identificadores de sesión definida como sigue:

$$\mathcal{I}([n]_s) = s$$

$$\mathcal{I}([v_1, \ldots, v_k{}_{v_0}]_s) = s$$

Semántica operacional III

Semántica operacional III

Ahora sí, pasemos a la definición de la relación de reducción.

Semántica operacional III

Ahora sí, pasemos a la definición de la relación de reducción.

$$(\text{Com}) \quad \frac{\wedge_{i=1}^{j} V_i \stackrel{f}{=} V_i'}{\langle V_1, \ldots, V_k \rangle.\mathcal{P} \mid (V_1', \ldots, V_j'; x_{j+1}, \ldots, x_k).\mathcal{P}' \rightarrow_{\mathcal{R}} \mathcal{P} \mid \mathcal{P}'[V_{j+1}'/x_{j+1}, \ldots, V_k'/x_k]}$$

$$(\text{Dec}) \quad \frac{\wedge_{i=0}^{j} V_i \stackrel{f}{=} V_i' \ \wedge \ \vee_{i=1}^{j} \mathcal{R}(\mathcal{I}(V_i), \mathcal{I}(V_i'))}{\text{decrypt } \{V_1, \ldots, V_k\}_{V_0} \text{ as } \{V_1', \ldots, V_j'; x_{j+1}, \ldots, x_k\}_{V_0'}^{f} \text{ in } \mathcal{P} \rightarrow_{\mathcal{R}} \mathcal{P}[V_{j+1}'/x_{j+1}, \ldots, V_k'/x_k]}$$

$$(\text{Res}) \quad \frac{\mathcal{P} \rightarrow_{\mathcal{R}} \mathcal{P}'}{(\nu\ [n]_s)\mathcal{P} \rightarrow_{\mathcal{R}} (\nu\ [n]_s)\mathcal{P}'} \qquad (\text{Repl}) \quad [!P]_s \rightarrow_{\mathcal{R}} \mathcal{T}(P, s) \mid [!P]_{s'} \quad (s' \text{ is fresh})$$

$$(\text{Par}) \quad \frac{\mathcal{P}_1 \rightarrow_{\mathcal{R}} \mathcal{P}_1'}{\mathcal{P}_1 \mid \mathcal{P}_2 \rightarrow_{\mathcal{R}} \mathcal{P}_1' \mid \mathcal{P}_2} \qquad (\text{Congr}) \quad \frac{P \equiv P' \ \wedge \ \mathcal{T}(P', s) \rightarrow_{\mathcal{R}} \mathcal{T}(P'', s)}{\mathcal{T}(P, s) \rightarrow_{\mathcal{R}} \mathcal{T}(P'', s)}$$

Semántica operacional IV

Semántica operacional IV

Con la relación de reducción definida podemos pasar a dar una de las definiciones más relevantes: la de frescura.

Semántica operacional IV

Con la relación de reducción definida podemos pasar a dar una de las definiciones más relevantes: la de frescura.

Def. Frescura. Un proceso P asegura la propiedad de frescura si, para todas las ejecuciones posibles $P \rightarrow^*_{\mathcal{R}} P' \rightarrow P''$ cuando $P' \rightarrow P''$ se deriva usando (Dec) en

$$\text{decrypt } [\{V_1, \ldots, V_k\}_{V_0}]_s \text{ as } \left\{V'_1, \ldots, V'_j; x_{j+1}, \ldots, x_k\right\}^{I}_{V'} \text{ in } P,$$

existe al menos un i $(1 \leq i \leq j)$ tal que $\mathcal{I}(V_i) = \mathcal{I}(V'_i)$.

Ejemplo: Protocolo Wide Mouthed Frog

Ejemplo: Protocolo Wide Mouthed Frog

Se usa una versión simplificada (sin timestamps) del protocolo WMF, un protocolo de gestión de claves simétrico cuyo objetivo es establecer un secreto clave de sesión K_{ab} entre A y B que comparten sus claves secretas K_A y K_B con un servidor de confianza S, para mostrar como ejemplo.

Ejemplo: Protocolo Wide Mouthed Frog

Se usa una versión simplificada (sin timestamps) del protocolo WMF, un protocolo de gestión de claves simétrico cuyo objetivo es establecer un secreto clave de sesión K_{ab} entre A y B que comparten sus claves secretas K_A y K_B con un servidor de confianza S, para mostrar como ejemplo.

Narración:

$$1.\ A \rightarrow S : \{B, K_{ab}\}_{K_A}$$
$$2.\ S \rightarrow B : \{A, K_{ab}\}_{K_B}$$
$$3.\ B \rightarrow A : \{Msg\}_{K_{ab}}$$

Ejemplo: protocolo Wide Mouthed Frog

Ejemplo: protocolo Wide Mouthed Frog

Especificación LySA:

Ejemplo: protocolo Wide Mouthed Frog

Especificación LySA:

$$
\begin{array}{lll}
1. & A & (\nu\ K_{ab}) \\
 & A \rightarrow & \langle A, S, \{B, K_{ab}\}_{K_A}\rangle. \\
3'. & \rightarrow A & (B, A; z). \\
3''. & A & \text{decrypt } z \text{ as } \{; z_m\}^{l1}_{K_{ab}} \text{ in } 0 \\
2'. & \rightarrow B \quad | & (S, B; y). \\
2''. & B & \text{decrypt } y \text{ as } \{A; k\}^{l2}_{K_B} \text{ in} \\
3. & B & (\nu\ Msg) \\
 & B \rightarrow & \langle B, A, \{Msg\}_k\rangle.0 \\
1'. & \rightarrow S \quad | & (A, S; p). \\
1''. & S & \text{decrypt } p \text{ as } \{B; k'\}^{l3}_{K_A} \text{ in} \\
2. & S \rightarrow & \langle S, B, \{A, k'\}_{K_B}\rangle.0
\end{array}
$$

Análisis de términos I

Análisis de términos I

- $\rho : X \to \wp(\text{Val})$ es el entorno de variables que asigna las variables a los conjuntos de valores a los que pueden estar vinculadas.

Análisis de términos I

- $\rho : X \to \wp(\text{Val})$ es el entorno de variables que asigna las variables a los conjuntos de valores a los que pueden estar vinculadas.
- Se utilizará $\rho \vdash \mathcal{E} : \vartheta$ para indicar que el conjunto ϑ es una estimación aceptable (una sobreaproximación correcta) de los posibles valores a los que el término $\mathcal{E}$ puede evaluar en el entorno ρ.

Análisis de términos I

- $\rho : X \to \wp(\text{Val})$ es el entorno de variables que asigna las variables a los conjuntos de valores a los que pueden estar vinculadas.
- Se utilizará $\rho \vdash \mathcal{E} : \vartheta$ para indicar que el conjunto ϑ es una estimación aceptable (una sobreaproximación correcta) de los posibles valores a los que el término $\mathcal{E}$ puede evaluar en el entorno ρ.
- Se emplean dos tipos de pruebas de pertenencia: $V \in \vartheta$ para comprobar si V está en el conjunto ϑ y $V \propto \vartheta$ para probar si hay un valor V' en ϑ que es igual a V, ignorando las anotaciones.

Análisis de términos II

Análisis de términos II

$$(\text{Name})\ \frac{[n]_s \in \vartheta}{\rho \models [n]_s : \vartheta} \qquad (\text{Var})\ \frac{\rho(x) \subseteq \vartheta}{\rho \models x : \vartheta}$$

$$(\text{Enc})\ \frac{\begin{array}{c}\wedge_{i=0}^{k} \rho \models \mathcal{E}_i : \vartheta_i \ \wedge \\ \forall V_0, \ldots, V_k : \wedge_{i=0}^{k} V_i \in \vartheta_i \Rightarrow [\{V_1, \ldots, V_k\}_{V_0}]_s \in \vartheta\end{array}}{\rho \models [\{\mathcal{E}_1, \ldots, \mathcal{E}_k\}_{\mathcal{E}_0}]_s : \vartheta}$$

Análisis de procesos I

Análisis de procesos I

- $\kappa \subseteq \wp(\mathsf{Val}^*)$ es el entorno de red abstracto que incluye todas las tuplas que forman un mensaje que puede fluir en la red.

Análisis de procesos I

- $\kappa \subseteq \wp(\mathsf{Val}^*)$ es el entorno de red abstracto que incluye todas las tuplas que forman un mensaje que puede fluir en la red.
- ψ es un conjunto posiblemente vacío de componentes de error que recopila una sobreaproximación de violaciones de frescura. Un $l \in \psi$ significa que el valor vinculado después de un descifrado exitoso, marcado con la etiqueta l, viola las anotaciones de frescura y, por lo tanto, no está permitido.

Análisis de procesos I

- $\kappa \subseteq \wp(\mathsf{Val}^*)$ es el entorno de red abstracto que incluye todas las tuplas que forman un mensaje que puede fluir en la red.
- ψ es un conjunto posiblemente vacío de componentes de error que recopila una sobreaproximación de violaciones de frescura. Un $l \in \psi$ significa que el valor vinculado después de un descifrado exitoso, marcado con la etiqueta l, viola las anotaciones de frescura y, por lo tanto, no está permitido.
- Se utiliza el símbolo $\rho, \kappa \vDash_{\mathsf{RM}} \mathcal{P} : \psi$ para expresar que ρ, κ, y ψ son estimaciones de análisis válidas para el proceso $\mathcal{P}$.

Análisis de procesos II

Análisis de procesos II

$$\text{(Out)}\quad \frac{\begin{array}{c}\wedge_{i=1}^{k}\rho \models \mathcal{E}_i : \vartheta_i \wedge \\ \forall V_1,\ldots,V_k \wedge_{i=1}^{k} V_i \in \vartheta_i \Rightarrow \\ \langle V_1,\ldots,V_k\rangle \in \kappa \ \wedge \rho,\kappa \models_{\mathsf{RM}} \mathcal{P} : \psi\end{array}}{\rho,\kappa \models_{\mathsf{RM}} \langle \mathcal{E}_1,\ldots,\mathcal{E}_k\rangle.\mathcal{P} : \psi}$$

$$\text{(Inp)}\quad \frac{\begin{array}{c}\wedge_{i=1}^{j}\rho \models \mathcal{E}_i : \vartheta_i \wedge \\ \forall \langle V_1,\ldots,V_k\rangle \in \kappa : \wedge_{i=1}^{j} V_i \propto \vartheta_i \Rightarrow \\ \wedge_{i=j+1}^{k} V_i \in \rho(x_i) \ \wedge \rho,\kappa \models_{\mathsf{RM}} \mathcal{P} : \psi\end{array}}{\rho,\kappa \models_{\mathsf{RM}} (\mathcal{E}_1,\ldots,\mathcal{E}_j; x_{j+1},\ldots,x_k).\mathcal{P} : \psi}$$

$$\text{(Dec)}\quad \frac{\begin{array}{c}\rho \models \mathcal{E} : \vartheta \ \wedge \ \wedge_{i=0}^{j}\rho \models \mathcal{E}_i : \vartheta_i \wedge \\ \forall [\{V_1,\ldots,V_k\}_{V_0}]_s \in \vartheta : \wedge_{i=0}^{j} V_i \propto \vartheta_i \Rightarrow \\ (\wedge_{i=j+1}^{k} V_i \in \rho(x_i) \ \wedge \rho,\kappa \models_{\mathsf{RM}} \mathcal{P} : \psi \ \wedge \\ (\nexists i : 1 \leq i \leq k : (\mathcal{I}(V_i) = \mathcal{I}(\mathcal{E}_i)) \Rightarrow l \in \psi))\end{array}}{\rho,\kappa \models_{\mathsf{RM}} \mathsf{decrypt}\ \mathcal{E}\ \mathsf{as}\ \{\mathcal{E}_1,\ldots,\mathcal{E}_j; x_{j+1},\ldots,x_k\}_{\mathcal{E}_0}^{l}\ \mathsf{in}\ \mathcal{P} : \psi}$$

Análisis de procesos III

Análisis de procesos III

$$\text{(Rep)}\ \frac{\rho, \kappa \models_{\mathsf{RM}} \mathcal{T}([P]_s) : \psi \ \wedge\ \rho, \kappa \models_{\mathsf{RM}} \mathcal{T}([P]_{s'}) : \psi}{\rho, \kappa \models_{\mathsf{RM}} [!P]_s : \psi} \qquad \text{(Nil)}\ \ \rho, \kappa \models_{\mathsf{RM}} 0 : \psi$$

$$\text{(Par)}\ \frac{\rho, \kappa \models_{\mathsf{RM}} \mathcal{P} : \psi \ \wedge\ \rho, \kappa \models_{\mathsf{RM}} \mathcal{Q} : \psi}{\rho, \kappa \models_{\mathsf{RM}} \mathcal{P} \mid \mathcal{Q} : \psi} \qquad \text{(Res)}\ \frac{\rho, \kappa \models_{\mathsf{RM}} \mathcal{P} : \psi}{\rho, \kappa \models_{\mathsf{RM}} (\nu [n]_s)\mathcal{P} : \psi}$$

Propiedades I

Propiedades I

Las estimaciones son resistentes a la sustitución de términos cerrados por variables.

Propiedades I

Las estimaciones son resistentes a la sustitución de términos cerrados por variables.

Lemma 1. ***(Substitution)***

1. $\rho \models \mathcal{E} : \vartheta$ *and* $\mathcal{E}' \in \rho(x)$ *imply* $\rho \models \mathcal{E}[\mathcal{E}'/x] : \vartheta$
2. $\rho, \kappa \models P : \psi$ *and* $\mathcal{E} \in \rho(x)$ *imply* $\rho, \kappa \models P[\mathcal{E}/x] : \psi$

Propiedades II

Propiedades II

Una estimación para un proceso extendido P es válida para cualquier proceso extendido congruente con P.

Propiedades II

Una estimación para un proceso extendido P es válida para cualquier proceso extendido congruente con P.

Lemma 2. *(Congruence)*
If $P \equiv Q$ *and* $\rho, \kappa \models \mathcal{T}([P]_s) : \psi$ *then* $\rho, \kappa \models \mathcal{T}([Q]_s) : \psi$

Propiedades III

Propiedades III

El resultado del análisis para un proceso es válido para sus derivados en la reducción $\mathcal{R}$.

Propiedades III

El resultado del análisis para un proceso es válido para sus derivados en la reducción $\mathcal{R}$.

Theorem 1. ***(Subject reduction)***

1. *If* $\mathcal{P} \rightarrow_{\mathcal{R}} \mathcal{Q}$ *and* $\rho, \kappa \models \mathcal{P} : \psi$ *then also* $\rho, \kappa \models \mathcal{Q} : \psi$;
2. *Furthermore, if* $\psi = \emptyset$ *then* $\mathcal{P} \rightarrow_{\mathsf{RM}} \mathcal{Q}$

Proof. The proof is done by induction of the inference of $\mathcal{P} \rightarrow_{\mathcal{R}} \mathcal{Q}$.

Propiedades IV

Propiedades IV

Si el conjunto de etiquetas ψ es vacío, entonces el monitor de referencia no puede abortar el proceso $\mathcal{P}$, ie.

$$\nexists \mathcal{Q}, \mathcal{Q}' / \mathcal{P} \rightarrow_{\mathcal{R}}^{*} \mathcal{Q} \rightarrow_{\mathsf{RM}} \mathcal{Q}' \wedge \mathcal{P} \rightarrow_{\mathcal{R}}^{*} \mathcal{Q} \nrightarrow_{\mathsf{RM}}$$

Propiedades IV

Si el conjunto de etiquetas ψ es vacío, entonces el monitor de referencia no puede abortar el proceso $\mathcal{P}$, ie.

$$\nexists \mathcal{Q}, \mathcal{Q}' / \mathcal{P} \rightarrow_{\mathcal{R}}^{*} \mathcal{Q} \rightarrow_{\mathsf{RM}} \mathcal{Q}' \wedge \mathcal{P} \rightarrow_{\mathcal{R}}^{*} \mathcal{Q} \nrightarrow_{\mathsf{RM}}$$

Theorem 2. ***(Static check for reference monitor)***

If $\rho, \kappa \models \mathcal{P} : \emptyset$ *then RM cannot abort* $\mathcal{P}$.

Proof Suppose *per absurdum* that such $\mathcal{Q}$ and $\mathcal{Q}'$ exist. A straightforward induction extends the subject reduction result to $\mathcal{P} \rightarrow^{*} \mathcal{Q}$ giving $\rho, \kappa \models_{\mathsf{RM}} \mathcal{Q} : \emptyset$. Theorem 1 part 2 of applied to $\mathcal{Q} \rightarrow \mathcal{Q}'$ gives $\mathcal{Q} \rightarrow_{\mathsf{RM}} \mathcal{Q}'$ which is a contradiction.

Ejemplo: análisis estático de WMF

Ejemplo: análisis estático de WMF

Análisis:

Ejemplo: análisis estático de WMF

Análisis:

$$\rho, \kappa \models_{\mathsf{RM}} WMF : \psi$$

where ρ, κ and ψ have the following entries

$$\begin{aligned}
\rho : \; & y \mapsto \{\{[A]_0, [K_{ab}]_0\}_{[K_B]_0}, \{[A]_1, [K_{ab}]_1\}_{[K_B]_1}\} \\
& z \mapsto \{\{[Msg]_0\}_{[K_{ab}]_0}, \{[Msg]_1\}_{[K_{ab}]_1}\} \\
& p \mapsto \{\{[B]_0, [K_{ab}]_0\}_{[K_A]_0}, \{[B]_1, [K_{ab}]_1\}_{[K_A]_1}\} \\
& k \mapsto \{[K_{ab}]_0, [K_{ab}]_1\} \\
& k' \mapsto \{[K_{ab}]_0, [K_{ab}]_1\} \\
& z_m \mapsto \{[Msg]_0, [Msg]_1\}
\end{aligned}$$

$$\begin{aligned}
\kappa : \; & \{\langle [A]_0, [S]_0, [\{[B]_0, [K_{ab}]_0\}_{[K_A]_0}]_0\rangle, \langle [A]_1, [S]_1, [\{[B]_1, [K_{ab}]_1\}_{[K_A]_1}]_1\rangle\} \cup \\
& \{\langle [B]_0, [A]_0, [\{[Msg]_0\}_{[K_{ab}]_0}]_0\rangle, \langle [B]_1, [A]_1, [\{[Msg]_1\}_{[K_{ab}]_1}]_1\rangle\} \cup \\
& \{\langle [S]_0, [B]_0, [\{[A]_0, [K_{ab}]_0\}_{[K_B]_0}]_0\rangle, \langle [S]_1, [B]_1, [\{[A]_1, [K_{ab}]_1\}_{[K_B]_1}]_1\rangle\}
\end{aligned}$$

$$\psi : \{l1, l2, l3\}$$

Ejemplo: análisis estático de WMF

Ejemplo: análisis estático de WMF

Posible ataque:

Ejemplo: análisis estático de WMF

Posible ataque:

$$
\begin{array}{rrl}
1. & [A]_1 & \rightarrow [S]_1 : \{[B]_1, [K_{ab}]_1\}_{[K_A]_1} \\
2. & [S]_1 & \rightarrow M : \{[A]_1, [K_{ab}]_1\}_{[K_B]_1} \\
 & M & \rightarrow [B]_1 : \{[A]_0, [K_{ab}]_0\}_{[K_B]_0} \\
3. & [B]_1 & \rightarrow [A]_1 : \{[Msg]_1\}_{[K_{ab}]_0}
\end{array}
$$

Modelado de atacantes I

Modelado de atacantes I

- Se dice que un proceso $\mathcal{P}_{\text{sys}}$ tiene el tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\text{Enc}})$ cuando:
  1. Es cerrado.
  2. Todos los nombres libres de $\mathcal{P}_{\text{sys}}$ están en $\mathcal{N}_f$.
  3. Todas las aridades utilizadas para enviar o recibir están en $\mathcal{A}_\kappa$.
  4. Todas las aridades utilizadas para encriptar o desencriptar están en $\mathcal{A}_{\text{Enc}}$.

Modelado de atacantes I

- Se dice que un proceso $\mathcal{P}_{\mathsf{sys}}$ tiene el tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$ cuando:
  1. Es cerrado.
  2. Todos los nombres libres de $\mathcal{P}_{\mathsf{sys}}$ están en $\mathcal{N}_f$.
  3. Todas las aridades utilizadas para enviar o recibir están en $\mathcal{A}_\kappa$.
  4. Todas las aridades utilizadas para encriptar o desencriptar están en $\mathcal{A}_{\mathsf{Enc}}$.
- Se considerarán atacantes Dolev-Yao activos.

Modelando atacantes II

Modelando atacantes II

Atacantes Dolev-Yao activos:

Modelando atacantes II

Atacantes Dolev-Yao activos:

(1) $\wedge_{k \in \mathcal{A}_\kappa} \ \forall \ \langle v_1, \ldots, v_k \rangle \in \kappa : \ \wedge_{i=1}^{k} v_i \in \rho(z_\bullet)$
the attacker may learn by eavesdropping

(2) $\wedge_{k \in \mathcal{A}_{\mathsf{Enc}}} \ \forall [\{v_1, \ldots, v_k\}_{v_0}]_s \in \rho(z_\bullet) :$
$v_0 \propto \rho(z_\bullet) \Rightarrow \wedge_{i=1}^{k} v_i \in \rho(z_\bullet)$
the attacker may learn by decrypting messages with keys already known

(3) $\wedge_{k \in \mathcal{A}_{\mathsf{Enc}}} \ \forall \ v_0, \ldots, v_k : \wedge_{i=0}^{k} v_i \in \rho(z_\bullet) \Rightarrow [\{v_1, \ldots, v_k\}_{v_0}]_{s_\bullet} \in \rho(z_\bullet)$
the attacker may construct new encryptions using the keys known

(4) $\wedge_{k \in \mathcal{A}_\kappa} \ \forall v_1, \ldots, v_k : \wedge_{i=1}^{k} v_i \in \rho(z_\bullet) \Rightarrow \langle v_1, \ldots, v_k \rangle \in \kappa$
the attacker may actively forge new communications

(5) $\{[n_\bullet]_{s_\bullet}\} \cup \mathcal{N}_f \subseteq \rho(z_\bullet)$
the attacker initially has some knowledge

Modelando atacantes III

Modelando atacantes III

Se define una fórmula $\mathcal{F}_{\mathsf{RM}}^{\mathsf{DY}}$ del tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$ como la conjunción de los cinco componentes en la tabla mostrada anteriormente

Modelando atacantes III

Se define una fórmula $\mathcal{F}_{\mathsf{RM}}^{\mathsf{DY}}$ del tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$ como la conjunción de los cinco componentes en la tabla mostrada anteriormente
y se establece que la fórmula $\mathcal{F}_{\mathsf{RM}}^{\mathsf{DY}}$ es capaz de caracterizar el efecto potencial de todos los atacantes $\mathcal{Q}$ del tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$.

Modelando atacantes III

Se define una fórmula $\mathcal{F}^{\mathsf{DY}}_{\mathsf{RM}}$ del tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$ como la conjunción de los cinco componentes en la tabla mostrada anteriormente
y se establece que la fórmula $\mathcal{F}^{\mathsf{DY}}_{\mathsf{RM}}$ es capaz de caracterizar el efecto potencial de todos los atacantes $\mathcal{Q}$ del tipo $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$.

Theorem 3. ***(Correctness of the extended Dolev-Yao condition)***

If (ρ, κ) satisfies $\mathcal{F}^{\mathsf{DY}}_{\mathsf{RM}}$ of type $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$ then there exists ψ such that for all attackers $\mathcal{Q}$ of type $(\mathcal{N}_f, \mathcal{A}_\kappa, \mathcal{A}_{\mathsf{Enc}})$ $\rho, \kappa \models_{\mathsf{RM}} \overline{\mathcal{Q}} : \psi$

Proof. The proof is done by structural induction on $\overline{\mathcal{Q}}$.

Índice

Frescura dinámica

Frescura dinámica

Se dice que $\mathcal{P}_{\text{sys}}$ garantiza frescura dinámica con respecto a las anotaciones en $\mathcal{P}_{\text{sys}}$ si el monitor de referencia $\mathcal{RM}$ no puede abortar $\mathcal{P}_{\text{sys}} \mid \mathcal{Q}$ independientemente de la elección del atacante $\mathcal{Q}$.

Frescura dinámica

Se dice que $\mathcal{P}_{\text{sys}}$ garantiza frescura dinámica con respecto a las anotaciones en $\mathcal{P}_{\text{sys}}$ si el monitor de referencia $\mathcal{RM}$ no puede abortar $\mathcal{P}_{\text{sys}} \mid \mathcal{Q}$ independientemente de la elección del atacante $\mathcal{Q}$.
Se muestra que frescura estática implica frescura dinámica.

Frescura dinámica

Se dice que $\mathcal{P}_{\mathsf{sys}}$ garantiza frescura dinámica con respecto a las anotaciones en $\mathcal{P}_{\mathsf{sys}}$ si el monitor de referencia $\mathcal{RM}$ no puede abortar $\mathcal{P}_{\mathsf{sys}} \mid \mathcal{Q}$ independientemente de la elección del atacante $\mathcal{Q}$.
Se muestra que frescura estática implica frescura dinámica.

Theorem 4. *If $\mathcal{P}$ guarantees static freshness then $\mathcal{P}$ guarantees dynamic freshness.*

Proof. If $\rho, \kappa \models_{\mathsf{RM}} \mathcal{P}_{sys} : \emptyset$ and (ρ, κ) satisfies $\mathcal{F}^{\mathsf{DY}}_{\mathsf{RM}}$ then, by Theorems 2 and 3, RM does not abort $\mathcal{P}_{sys} \mid \overline{\mathcal{Q}}$ regardless of the choice of attacker $\mathcal{Q}$.

Implementación

Implementación

Para obtener una implementación, se transforma el análisis en una formación lógicamente equivalente escrita en Alternation-free Least Fixed Point logic (ALFP) (Nielson-Seidl-Nielson, 2002), y se utiliza el Succinct Solver (Nielson-Seidl-Nielson, 2002), que calcula la interpretación mínima de los símbolos predicados en una fórmula ALFP dada.

Validación del protocolo de Needham-Schroeder I

Validación del protocolo de Needham-Schroeder I

1. $A \rightarrow S : A, B, N_a$
2. $S \rightarrow A : \{N_a, B, K, \{K, A\}_{K_b}\}_{K_a}$
3. $A \rightarrow B : \{A, K\}_{K_b}$
4. $B \rightarrow A : \{N_b\}_K$
5. $A \rightarrow B : \{N_b - 1\}_K$
6. $A \rightarrow B : \{Msg\}_K$

the protocol narration

1. $A \rightarrow S : A, B, N_a$
2. $S \rightarrow A : \{N_a, B, K, \{K, A\}_{K_b}\}_{K_a}$
3. $M(A) \rightarrow B : \{A, K'\}_{K_b}$
4. $B \rightarrow M(A) : \{N_b\}_{K'}$
5. $M(A) \rightarrow B : \{N_b - 1\}_{K'}$
6. $M(A) \rightarrow B : \{Msg\}_{K'}$

a replay attack scenario

Validación del protocolo de Needham-Schroeder II

Validación del protocolo de Needham-Schroeder II

La solución propuesta por Needham y Schroeder implica la introducción de un nuevo fresco llamado N'_a. Después de la corrección, el protocolo incluirá una solicitud adicional del nuevo valor N'_a por parte de A a B, y este valor se enviará al servidor para su retorno encriptado.
En el nuevo protocolo corregido, las primeras tres etapas del intercambio de claves se modifican como sigue:

$$
\begin{aligned}
&1.\ A \rightarrow S: && A, B, N_a, N'_a \\
&2.\ S \rightarrow A: && \{N_a, B, K, \{A, N'_a, K\}_{K_b}\}_{K_a} \\
&3.\ M(A) \rightarrow B: && \{A, N'_a, K\}_{K_b}
\end{aligned}
$$

Después de aplicar el análisis al protocolo corregido, el resultado indica que no hay violaciones posibles, es decir, $\psi = \emptyset$.

Índice

Comentarios finales

Comentarios finales

- Trabajos relacionados: Authenticity by Tagging and Typing (Bugliesi-Focardi-Maffei, 2004) - Types and Effects for Asymmetric Cryptographic Protocols (Gordon-Jeffrey, 2002).

Comentarios finales

- Trabajos relacionados: Authenticity by Tagging and Typing (Bugliesi-Focardi-Maffei, 2004) - Types and Effects for Asymmetric Cryptographic Protocols (Gordon-Jeffrey, 2002).
- Este trabajo está enmarcado en un proyecto donde varias propiedades de comunicación de protocolos son analizadas mediante anotaciones y fácilmente se puede combinar con otro tipo de anotaciones, por ejemplo anotaciones de confidencialidad o anotaciones para el abordaje de type flaw attacks.