Logika BAN

Piotr Iwaniuk

14 grudnia 2011

Plan

Plan

Czym jest logika BAN?

- Rodzaj logiki do opisywania i weryfikacji protokołów uwierzytelniania.
- Operuje na powszechnie używanych opisach protokołów.
- Jest rozstrzygalna.

Motywacje powstania

- Powstało wiele protokołów uwierzytelniania, lub przesyłania zaszyfrowanych informacji.
- Wiele z nich powiela istniejące schematy nie wiedząc czy są dobre.
- Chcemy wiedzieć czy protokoły spełniają swoje założenia.
- Chcemy wiedzieć czy przesyłane są nadmiarowe dane.

Burrows, Abadi, Needham

Roger Needham: protokoły dystrybucji symetrycznych i asymetrycznych kluczy.

Martin Abadi: Theory of Objects

Michael Burrows: Burrows-Wheeler transform (bzip2), AltaVista

Plan

Elementy modelu

- Korzystamy z logiki wielorodzajowej.
- Mamy takie elementy jak: agenci, klucze, formuły/zdania.
- Klucze mogą być publiczne, prywatne i symetryczne.

Podstawowe konstrukty

P believes X

P jest przekonany o prawdziwości X.

P sees X

P otrzymał od wiadomość zawierającą X.

P said X

P przesłał X w przeszłości.

Podstawowe konstrukty

P controls X

X jest w kompetencji P.

fresh(X)

X nie było przesyłane we wcześniejszych sesjach protokołu.

Podstawowe konstrukty

$P \overset{K}{\leftrightarrow} Q$

P i Q używają klucza symetrycznego K.

$\overset{K}{\mapsto} P$

P ma klucz publiczny K.

$P \overset{K}{\rightleftharpoons} Q$

X jest formułą znaną tylko P i Q.

Podstawowe konstrukty

$\{X\}_K$

X jest zaszyfrowany kluczem K.

$\langle X \rangle_Y$

X jest poświadczone przez Y.

Postulaty

Postulaty związane ze znaczeniem komunikatu:

Postulat dla kluczy symetrycznych

$$\frac{P \textbf{ believes } Q \overset{K}{\leftrightarrow} P, \quad P \textbf{ sees } \{X\}_K}{P \textbf{ believes } Q \textbf{ said } X}$$

Postulat dla kluczy asymetrycznych

$$\frac{P \textbf{ believes } \overset{K}{\mapsto} Q, \quad P \textbf{ sees } \{X\}_{K^{-1}}}{P \textbf{ believes } Q \textbf{ said } X}$$

Postulat dla współdzielonych sekretów

$$\frac{P \textbf{ believes } Q \overset{K}{\rightleftharpoons} P, \quad P \textbf{ sees } \langle X \rangle_Y}{P \textbf{ believes } Q \textbf{ said } X}$$

Postulaty

Zasada weryfikacji nonce'a

$$\frac{P \textbf{ believes fresh } X, \quad P \textbf{ believes } Q \textbf{ said } X}{P \textbf{ believes } Q \textbf{ believes } X}$$

Zasada kontroli

$$\frac{P \textbf{ believes } Q \textbf{ controls } X, \quad P \textbf{ believes } Q \textbf{ believes } X}{P \textbf{ believes } X}$$

Postulaty

Odpakowywanie

$$\frac{P \text{ sees } \langle X \rangle_Y}{P \text{ sees } X}, \quad \frac{P \text{ sees } (X, Y)}{P \text{ sees } X},$$

$$\frac{P \text{ believes } Q \overset{K}{\leftrightarrow} P, \quad P \text{ sees } \{X\}_K}{P \text{ sees } X},$$

$$\frac{P \text{ believes } \overset{K}{\mapsto} P, \quad P \text{ sees } \{X\}_K}{P \text{ sees } X},$$

$$\frac{P \text{ believes } \overset{K}{\mapsto} Q, \quad P \text{ sees } \{X\}_{K^{-1}}}{P \text{ sees } X}$$

Postulaty

Aktualność

$$\frac{P \text{ believes } \mathbf{fresh}(X)}{P \text{ believes } \mathbf{fresh}(X, Y)}$$

Kwantyfikacja

Zapis:

$$A \textbf{ believes } S \textbf{ controls } A \overset{K}{\leftrightarrow} B$$

jest równoważny:

$$A \textbf{ believes } \forall_K (S \textbf{ controls } A \overset{K}{\leftrightarrow} B)$$

W bardziej skomplikowanych formułach konieczne jest umieszczanie kwantyfikatorów, aby określić ich zakres. Przykładowo poniższe dwa zdania nie są równoważne:

$$A \textbf{ believes } \forall_K (S \textbf{ controls } B \textbf{ controls } A \overset{K}{\leftrightarrow} B)$$

$$A \textbf{ believes } S \textbf{ controls } \forall_K (B \textbf{ controls } A \overset{K}{\leftrightarrow} B)$$

Implementacyjny zapis protokołu

Najczęściej protokoły opisuje się przy pomocy listy składającej się trójek:

$P \rightarrow Q$: wiadomość

Taki sposób zapisu oddaje dokładnie przesyłane dane, ale nie rozróżnia znaczenia komunikatów.

Wyidealizowany zapis protokołu

W celu weryfikacji opis protokołu przekształca się do formy wyidealizowanej. Każda wiadomość jest reprezentowana przez formułę, np.:

Wiadomość:

$$A \rightarrow B : \{A, K_{AB}\}_{K_{BS}} \tag{1}$$

może być przekształcona na:

$$A \rightarrow B : \{A \overset{K_{AB}}{\leftrightarrow} B\}_{K_{BS}} \tag{2}$$

Analiza protokołu

W celu analizy dodajemy adnotacje do wyidealizowanego zapisu protokołu podobnie jak w logice Hoare'a. Adnotacje dodaje się wg dwóch głównych zasad:

- Jeśli X zachodzi przed komunikatem $P \longrightarrow Q : Y$, to po komunikacie zachodzą X i Q **sees** Y.
- Jeśli Y wynika z X zgodnie z postulatami, to Y zachodzi tam gdzie zachodzi X.

Plan

Sformalizowanie uwierzytelniania

- Zazwyczaj na początku dodajemy standardowe założenia dotyczące wykorzystania protokołu (np. którzy agenci są zaufani).
- Z tych założeń chcemy wywnioskować interesujące nas właściwości.

$$A \textbf{ believes } A \overset{K}{\leftrightarrow} B, B \textbf{ believes } B \overset{K}{\leftrightarrow} A$$

$$A \textbf{ believes } B \textbf{ believes } A \overset{K}{\leftrightarrow} B, B \textbf{ believes } A \textbf{ believes } B \overset{K}{\leftrightarrow} A$$

Table I. Summary of Results

	Needham-Schroeder shared key	Otway-Rees	Kerberos	Wide-mouthed frog	Yahalom	Andrew RPC	Needham-Schroeder public key	CCITT X.509
Goal	Distribute key	Distribute key	Distribute key	Distribute key[a]	Distribute key	Distribute extra key	Establish secrets	Transfer data
Keys	Shared	Shared	Shared	Shared	Shared	Shared	Public	Public
Uses secrets					×		×	
Nonces/clocks	Nonces	Nonces	Clocks	Clocks	Nonces	Nonces	Nonces	Both
Proves presence of	*A* and *B*	*B*	*A* and *B*[b]	*A*	*A* and *B*	*A* and *B*	*A* and *B*	*A* and *B*[b]
Redundancy	×	×	×		×	×		×
Bugs	×					×	×	×[c]

[a] In this case, *A*, rather than a trusted server, generates the key.
[b] *B*'s presence is guaranteed to *A* only if optimal protocol steps are used.
[c] Security breaches do not even require key compromise.

Plan

Podsumowanie

- Udało się znaleźć błędy i niepotrzebne elementy w kilku różnych protokołach bezpieczeństwa.
- Autorzy artykułu opracowali wyspecjalizowany formalizm do weryfikacji protokołów.
- Niektóre sformułowania użyte w artykule są nieprecyzyjne.

Ciąg dalszy nastąpi (?)

Literatura

Michael Burrows, Martín Abadi, and Roger Needham.
A logic of authentication.
ACM TRANSACTIONS ON COMPUTER SYSTEMS, 8:18–36, 1990.