Protokollverifikation mit BAN-Logik

Maximilian Schlund
Fakultät für Mathematik, TU München
schlund@in.tum.de

***Zusammenfassung*—Wir geben eine kurze Einführung in BAN-Logik und ihre Anwendung zur Protokollverifikation. Diskutiert werden auch Probleme und Beschränkungen dieser Methode**

***Index Terms*—BAN-Logik, Verifikation, Kryptographische Protokolle**

I. Motivation

DAS sichere Design kryptographischer Protokollen zur Authentifizierung von Kommunikationsteilnehmern ist hoch komplex und oft unterlaufen den Entwicklern dabei subtile aber dennoch fatale Fehler. Als Beispiel sei hier das Public-Key-Needham-Schroeder Protokoll [NS78] genannt auf das Lowe 17 Jahre nach seiner Publikation einen Man-in-the-Middle Angriff fand [Low95]. Auch der erste Entwurf des X.509 Standards enthielt diverse schwere Sicherheitsmängel, wie beispielsweise in [BAN90] gezeigt wird. Als ein weiteres Beispiel ist das bekannte symmetrische Needham-Schroeder-Protokoll zu nennen:

M1: $A \to S : A, B, N_a$
M2: $S \to A : \{N_a, B, K_{ab}, \{K_{ab}, A\}_{K_{bs}}\}_{K_{as}}$
M3: $A \to B : \{K_{ab}, A\}_{K_{bs}}$
M4: $B \to A : \{N_b\}_{K_{ab}}$
M5: $A \to B : \{N_b - 1\}_{K_{ab}}$

Hier ist im Schritt M3 ein Replay-Angriff durch einen Angreifer E möglich. Angenommen E hat Zugang zu einem alten Sitzungsschlüssel K'_{ab} so kann er die alte aufgezeichnete Nachricht M3 wiedereinspielen:

M3' $E(A) \to B : \{K'_{ab}, A\}_{K_{bs}}$
M4' $B \to E(A) : \{N_b\}_{K'_{ab}}$
M5' $E(A) \to B : \{N_b - 1\}_{K'_{ab}}$

Nach Ablauf dieser Schritte *glaubt* B mit A zu reden, redet in Wirklichkeit aber mit E. Eine Möglichkeit solche und ähnliche Fehler in Protokollen zu finden oder sogar ihre Abwesenheit zu beweisen sind mathematische Analysetechniken sog. *formale Methoden*. Hierunter fallen beispielsweise

- Model Checking (E. Clarke [MCJ97])
- Induktive Verifikation (L. C. Paulson [Pau00])
- Prozesskalküle (π-Kalkül, R. Milner - CCS [Mil82])
- Modallogik (*BAN-Logik*, GNY-Logik [GNY90])

Das grundsätzliche Vorgehen besteht immer darin, ein mathematisches Modell des Protokolls und der Ablaufumgebung (Kommunikationskanäle, Teilnehmer, Angreifer,...) zu erstellen und in diesem Modell bestimmte Sicherheitseigenschaften zu beweisen oder zu widerlegen. Wir werden hier den Ansatz der BAN-Logik näher betrachten.

II. BAN-Syntax

Die Sprache der BAN-Logik besteht aus Konstanten für Nachrichten, Teilnehmer ($A, B, S, \dots$), Schlüssel (K_{xy}) sowie aus diversen Funktionssymbolen $\models, \triangleleft, \mid\!\sim, \dots$. Die Formeln der BAN-Logik sind induktiv definiert: Jede Konstante ist eine Formel und falls X eine Formel und A, B und S Teilnehmerkonstanten und Y eine Nachrichtenkonstante ist, so sind auch die folgenden Konstrukte Formeln:

- $A \models X$: A *glaubt*/kennt X
- $A \triangleleft X$: A *sieht* X (d.h. A empfängt „Nachricht“ X)
- $A \mid\!\sim X$: A *sagte* X
- $S \Rightarrow X$: S kann bezüglich X *vertraut* werden
- $A \overset{K}{\longleftrightarrow} B$: K ist ein *gemeinsamer Schlüssel* (nur) für A und B
- $\overset{K}{\mapsto} A$: K ist öffentlicher Schlüssel von A
- $\sharp(X)$: Formel X ist *„frisch“* bzw. „aktuell“ (zur Modellierung von Timestamps und Nonces)
- $\{X\}_K$: X ist *verschlüsselt* mit K
- $A \overset{Y}{\rightleftharpoons} B$: Y ist *gemeinsames Geheimnis* von A und B
- $\langle X \rangle_Y$: X ist *kombiniert* mit Y (Y ist meist ein Geheimnis, das Herkunft von X bezeugt)

Aus der Definition ergibt sich natürlich, dass BAN-Formeln immer *rechts-assoziativ* zu lesen sind: $A \models B \models X = A \models (B \models X)$

Die wichtigsten Inferenzregeln der BAN-Logik sind die folgenden. Zuerst die „Message Meaning Rules“:

$$\frac{A \models A \overset{K}{\longleftrightarrow} B, \quad A \triangleleft \{X\}_K}{A \models B \mid\!\sim X} \ (MM_{sym})$$

Intuitiv bedeutet sie: Falls A eine Nachricht X empfängt und sie glaubt, dass nur A und B den Schlüssel K besitzen, so wird A danach glauben, dass die Nachricht X von B gesendet wurde. Ähnliche Regeln kann man auch für asymetrische Systeme und geteilte Geheimnisse angeben:

$$\frac{A \models \overset{K}{\mapsto} B, \quad A \triangleleft \{X\}_{K^{-1}}}{A \models B \mid\!\sim X} \ (MM_{asym})$$

$$\frac{A \models A \overset{Y}{\rightleftharpoons} B, \quad A \triangleleft \langle X \rangle_Y}{P \models B \mid\!\sim X} \ (MM_{sec})$$

Die nächste Regel ist die „Nonce Verification Rule“:

$$\frac{A \models \sharp(X), \quad A \models B \mid\!\sim X}{A \models B \models X} \ (NV)$$

Intuitiv: Wenn A denkt dass X aktuell/frisch ist, und er glaubt das B einmal X gesagt hat, dann kann er davon ausgehen, dass B immer noch X glaubt. Bemerkung: Hier wird davon ausgegangen, dass Teilnehmer „ehrlich“ (honest) sind, d.h. u.a. nur Nachrichten versenden welche sie

glauben. Die „Jurisdiction Rule“ modelliert die Autorität eines vertrauenswürdigen Teilnehmers:

$$\frac{A \models S \Rightarrow X, \quad A \models S \models X}{A \models X} \ (J)$$

Intuitiv: Wenn A S in Bezug auf X vertraut, so wird er auch dessen Glauben in X annehmen. Des weiteren benötigt man noch diverse Hilfsregeln ähnlich zu bekannten Introduktions und Eliminationsregeln wie dem Kalkül des natürlichen Schließens. Hier nur ein paar Beispiele, für andere Funktionssymbole kann man in kanonischer Weise meist ähnliche Regeln postulieren:

$$\frac{A \models \sharp(X)}{A \models \sharp(X, Y)} \ (F_I)$$

Intuitiv: Falls ein Teil einer Formel frisch ist, so ist die ganze Formel frisch.

$$\frac{A \models (X, Y)}{A \models X} \ (B_E)$$

Falls A eine Nachricht glaubt, so glaubt sie auch jede Komponente. Genauso sinnvoll ist:

$$\frac{A \models B \models (X, Y)}{A \models B \models X} \ (BB_E)$$

Wichtig: Diese Regeln sind rein **syntaktische** Ersetzungsregeln! Es ist überhaupt nicht klar, dass damit irgend etwas sinnvolles modelliert oder bewiesen werden kann!

Definition 1: Seien Γ und Φ Formelmengen der BAN-Logik, wir schreiben $\Gamma \vdash \Phi$ falls sich jedes $\varphi \in \Phi$ durch Anwenden von Schlussregeln aus Γ herleiten lässt

III. Protokollverifikation

Als ersten Modellierungsschritt geht man von der „informellen“ Protokollbeschreibung zu einem formaleren Modell eines „idealisierten Protokolls“ über, welches BAN-Formeln als „Nachrichten“ enthält. Das Problem ist, dass dieser Übergang nicht durch schematische Ersetzung von Konstrukten geschieht, sondern nur mit Intuition über das Protokoll funktioniert. Dabei kommt es leicht zu Modellierungsfehlern, die die Effektivität von BAN-Logik sofort zerstören, da oft fehlerhafte Protokolle zu fehlerfreien idealisiert werden, die dann als korrekt bewiesen werden können (siehe als Beispiel [BM93], [vO94]).

Definition 2: Ein „idealisiertes Protokoll“ ist eine Folge von „Send-Statements“ $S_1, S_2, \ldots, S_n$. Hierbei sind die S_i von der Form $P \to Q : X$, wobei P und Q Teilnehmer sind und X eine BAN-Formel ist.
Analog zum Vorgehen im Floyd-Hoare-Kalkül annotieren wir ein Protokoll mit Zusicherungen, d.h. Formeln in BAN-Logik.

Definition 3 ((lokale) Konsistenz)
$[A_{i-1}] \ \underbrace{P \to Q : X}_{S_i} \ [A_i]$ heißt *lokal konsistent annotiert*, gdw.

$$A_{i-1} \cup (Q \triangleleft X) \vdash A_i$$

Die Sequenz $[A_0]S_1[A_1]S_2 \ldots [A_{n-1}]S_n[A_n]$ heißt *konsistent annotiert* wenn jedes Tripel $[A_{i-1}]\ S_i\ [A_i]$ lokal konsistent ist.
Wichtig: Konsistenz selbst sagt nichts über die Korrektheit eines Protokolls aus!

Um ein Protokoll zu verifizieren bräuchten eine Definition der Semantik eines Send-Statements. Hier liegt eine der größten Schwächen der BAN- Logik. Ein dubioser Schritt ist es die Semantik eines Send-Statements als eine Transformation von Formelmengen zu sehen, die an die logischen Schlussregeln angelehnt sind. Das ist auch die Vorgehensweise der Autoren [BAN90], die dazu führt, dass die Modellierung mit BAN-Formeln sehr stark von der Realität abstrahiert und so eine weitere Quelle von gefährlichen Modellierungsfehlern darstellt, da die Semantik der Formeln nicht klar definiert wurde (vgl. [BM93], [vO94]). Wenig später haben Abadi und Tuttle die BAN-Logik leicht modifiziert [AT91] und mit einer formalen Semantik versehen, die viele Probleme ausräumt, jedoch nicht den Übergang von einem realen Protokolls zu einem idealisierten Protokoll vereinfacht.

IV. Beispiel

Wir betrachten noch einmal das Needham-Schroeder-Protokoll in idealisierter Form:

$$\text{M2'} \ \ S \to A : \left\{ N_a, A \overset{K_{ab}}{\longleftrightarrow} B, \sharp(A \overset{K_{ab}}{\longleftrightarrow} B), \left\{ A \overset{K_{ab}}{\longleftrightarrow} B \right\}_{K_{bs}} \right\}_{K_{as}}$$

$$\text{M3'} \ \ A \to B : \left\{ A \overset{K_{ab}}{\longleftrightarrow} B \right\}_{K_{bs}}$$

$$\text{M4'} \ \ B \to A : \left\{ N_b, A \overset{K_{ab}}{\longleftrightarrow} B \right\}_{K_{ab}}$$

$$\text{M5'} \ \ A \to B : \left\{ N_b, A \overset{K_{ab}}{\longleftrightarrow} B \right\}_{K_{ab}}$$

Ein Authentifikationsziel wäre beispielsweise zu beweisen, dass $B \models A \overset{K_{ab}}{\longleftrightarrow} B$, also dass B glaubt mit A einen gemeinsamen Schlüssel zu besitzen. Es stellt sich jedoch heraus, dass dieses Ziel nur dann ableitbar ist, wenn man die Annahme $B \models \sharp(A \overset{K_{ab}}{\longleftrightarrow} B)$ hinzufügt. Also B muss glauben, dass der Schlüssel den er erhält frisch ist. Das zeigt einen Fehler im Protokoll auf, denn diese Annahme ist praktisch nicht sinnvoll. Protokollfehler manifestieren sich in BAN-Beweisen also dadurch, dass die Ziele nur unter dubiosen Annahmen abgeleitet werden können.

Des weiteren benötigt man beim Beweis auch die Annahme, dass $B \models (S \Rightarrow \sharp(A \overset{K_{ab}}{\longleftrightarrow} B))$, also dass B dem Server S vertrauen muss, einen guten, neuen Schlüssel zu erzeugen. Die explizite Notwendigkeit solcher Annahmen führen oft zu einem besseren Verständnis der Mechanismen und Funktionsweisen eines Protokolls.

Des weiteren kann ein BAN-Beweis aufzeigen, dass bestimmte Nachrichten im Protokoll überflüssig sind. Dies ist genau dann der Fall, wenn die entsprechenden Formeln nicht zum Beweis der Authentifikationsziele benötigt werden.

V. Fazit und Ausblick

BAN-Logik ist ein einfaches Hilfsmittel um subtile Fehler im Protokollentwurf zu entdecken. Durch die formale

Herangehensweise werden implizite und intuitive Annahmen über Vertrauen und Glauben explizit formuliert und können mit der Realität abgeglichen werden. Ein Beweis in BAN-Logik hat jedoch nur Wert für die Realität, falls das modellierte Protokoll dem realen in seiner Funktionsweise entspricht. Hier liegt die größte Schwäche der Verifikation mit BAN-Logik:

- die Semantik von BAN-Logik ist in ihrer ursprünglichen Form unklar und informell. Die Korrekturen in [AT91] schaffen mehr Klarheit.
- die Idealisierung eines realen Protokolls geschieht auf eine intuitive und informelle Art und Weise

Beides führt dazu, dass Verifikation mit BAN-Logik auf einer Ebene geschieht, die stark vom realen Protokoll abstrahiert. Ein Beweis in BAN-Logik ist daher mit Vorsicht zu genießen. Da BAN-Logik aber sogar entscheidbar ist (siehe [Mon99]) ist, kann sie als Unterstützung beim Design wertvolle Einsichten liefern.

Literatur

[AT91] Martin Abadi and Mark Tuttle. A semantics for a logic of authentication. In *In Proceedings of the ACM Symposium of Principles of Distributed Computing*, pages 201–216. ACM Press, 1991.

[BAN90] Michael Burrows, Martin Abadi, and Roger Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8:18–36, 1990.

[BM93] Colin Boyd and Wenbo Mao. On a limitation of ban logic. pages 240–247. Springer-Verlag, 1993.

[GNY90] Li Gong, Roger Needham, and Raphael Yahalom. Reasoning about belief in cryptographic protocols. In *Proceedings 1990 IEEE Symposium on Research in Security and Privacy*, pages 234–248. IEEE Computer Society Press, 1990.

[Low95] Gavin Lowe. An attack on the needham-schroeder public-key authentication protocol. *Information Processing Letters*, 56:131–133, 1995.

[MCJ97] Will Marrero, Edmund Clarke, and Somesh Jha. Model checking for security protocols. Technical report, Carnegie Mellon University, 1997.

[Mil82] R. Milner. *A Calculus of Communicating Systems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1982.

[Mon99] David Monniaux. Decision procedures for the analysis of cryptographic protocols by logics of belief. In *In 12th Computer Security Foundations Workshop. IEEE*, pages 44–54. IEEE, 1999.

[NS78] Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Commun. ACM*, 21(12):993–999, 1978.

[Pau00] Lawrence C. Paulson. The inductive approach to verifying cryptographic protocols. *Journal of Computer Security*, 6:85–128, 2000.

[vO94] Paul C. van Oorschot. An alternate explanation of two ban-logic "failures". In *EUROCRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 443–447, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.