Prace Naukowe Wyższej Szkoły Pedagogicznej w Częstochowie
Matematyka VIII, Częstochowa 2000 - 2001

The Completeness Theorem for a Logic of Authentication

Mirosław Kurkowski

Abstract

In a distributed computational system, security depends heavily on the use of secure protocols such as authentication protocols. In the past few years, a lot of attention has been paid to the use of special logics to analyze cryptographic authentication protocols. Burrows, Abadi, Needham and others have proposed a few logics for the analysis of these protocols (called BAN-logics). These are specialized versions of modal logics of belief, with special constructs for expressing some of the central concepts used in authentication processes. These logics have revealed many subtleties and serious flaws in published and widely applied protocols. Unfortunately, they have also created some confusions, for example they are not complete.

In the present paper we introduce a new logic of authentication, which is a modification of the BAN logic. It provides a convenient formal language for specifying and reasoning about cryptographic protocols requirements. We also provide an axiomatic inference system, a model of computation and semantics. We present some important properties of our logic, first of all the Completeness Theorem.

1. Introduction

Authentication is the act of determining the identity of agents (such as persons, computers, or servers) in a computer system. Authentication usually plays an important role in secure system, since a principal controlling a resource must have some way of identyfying principal requesting access to the resource. Authentication typically depends on secrets, such as passwords or encryption keys, that one agent can reveal or somehow use to prove its identity to others. Before these secrets can be used, however, they must be distributed to the principals in some way. An authentication protocol is a description of how these secrets are distributed to agents, and how these secrets are used to determine agents identities.

One of the first, important authentication protocols is *the Needham - Schroeder protocol with public key* [NS78] given as follows:

1. $A \to B \quad A, B, \{N_A, A\}_{KB}$

2. $B \to A \quad B, A, \{N_A, N_B\}_{KA}$

3. $A \to B \quad A, B, \{N_B\}_{KB}$

where A, B denote agents, KA is a public key of A (similarly KB), N_A is a special number generated by the agent A and used only once (called *nonce*).

In the first step of the protocol A sends to B a message encrypted under KB containing nonce N_A and his identifier. The agent B is the only one who can decrypt this data. In the second step, B sends to A a message encrypted under KA containing nonce N_A and nonce N_B. In the third step, A sends to B a message encrypted under KA containing nonce N_B. When the protocol is done agents A and B should be convinced about their identity.

Unfortunately, this and others protocols are not perfect. Here is one of method of attack the protocol:

a1. $A \to C \quad A, C, \{N_A, A\}_{KC}$

b1. $C(A) \to B \quad A, B, \{N_A, A\}_{KB}$

b2. $B \to C(A) \quad B, A, \{N_A, N_B\}_{KA}$

a2. $C \to A \quad C, A, \{N_A, N_B\}_{KA}$

a3. $A \to C \quad A, C, \{N_B\}_{KC}$

b3. $C(A) \to B \quad A, B, \{N_B\}_{KB}$

When all this information has been sent the agent B is convinced about his communication with A, while he really communicates with C.

There is no method which can say that the protocol is good. It is still searched for such a method. One of the ways of investigation is constructing logics which search cryptographic protocol requirements. The purpose of these logics is to formalize our informal reasoning about authentication protocols and to explain these protocols and their differences. Burrows, Abadi, Needham and others have proposed logics specifically tailored to the analysis of authentication protocols [BAN89], [GNY90], [AT91], [S91], [SvO94], [SM93]. These logics provide languages for describing the beliefs of the parties involved in a protocol, and sets of inference rules that describe how the beliefs evolve as the result of communication. They include special construct for expressing many of the central concepts used in authentication. Unfortunately, they have also created some confusions, for example, they are not complete.

In this paper we reformulate the logic in several ways. In the section 2 we provide a new language and an axiomatic inference system. In the section 3 and 4 a model of computation and semantics are presented. In the section 5 we point out some important properties of our logic, first of all the Completeness Theorem. Our logic is complete, but all our modification were done not only to achieve this completeness.

2. Syntax

Now we present our modification of the logic of authentication

2.1. Language

We begin with a definition of our language. Remind that we are analyzing idealized protocols in which principals send to each other messages; we define a language of messages M and a language of formulas F.
We assume the existence of a set T of primitive terms containing a number of disjoint sets of constant symbols: $T = T_A \cup T_P \cup T_K$, where $T_A = \{a_1, a_2, \ldots, a_n\}$ is an alphabet, $T_P = \{P, Q, \ldots, R\}$ is a set of symbols representing principals, $T_K = \{K, L, \ldots, H\}$ is a set of symbols representing encryption keys.

We define messages and formulas in the logic by mutual induction. The language M of messages is the smallest language over T satisfying the following conditions:

1. if $\alpha \in F$, then $\alpha \in M$,

2. if $X = (m_1, m_2, \ldots, m_n)$ and $m_1, m_2, \ldots, m_n \in T$, then $X \in M$,

3. if $X \in M, K \in T_K, P \in T_P$ then $\{X^P\}_K \in M, \{X^P\}_K$ - the message X encrypted under the key K.

The language F of formulas is the smallest language satisfying the conditions:

1. if $X \in M, P \in T_P$, then $P \rhd X, P \lhd X \in F, P \rhd X - P$ has received the message $X, P \lhd X - P$ has sent the message X,

2. if $P \in T_P, K \in T_K$, then $\rightarrow^K P \in F, \rightarrow^K P - P$ has the key K;

3. if $\alpha, \beta \in F$, then $\neg\alpha, \alpha \wedge \beta, \alpha \vee \beta, \alpha \Rightarrow \beta, \alpha \equiv \beta \in F$.

4. if $P \in T_P, \alpha \in F$, then $P \mid\equiv \alpha \in F, P \mid\equiv \alpha - P$ believes that α is true.

2.2 Axiomatic inference system

Our axiom system includes only one inference rule: the Modus Ponens rule

$$r_0 : \frac{\alpha, \alpha \Rightarrow \beta}{\beta}.$$

The axioms are all substitutions of formulas from the set F into classical propositional tautologies (let A_0 be a set of all these substitutions) and the following axioms schemes:

A1. $P \mid\equiv \alpha_0$ for all $\alpha_0 \in A_0$,

A2. $P \mid\equiv \alpha \wedge P \mid\equiv (\alpha \Rightarrow \beta) \Rightarrow P \mid\equiv \beta$,

A3. $P \mid\equiv \alpha \Rightarrow P \mid\equiv (P \mid\equiv \alpha)$,

A4. $P \rhd X \Rightarrow P \mid\equiv (P \rhd X)$,

A5. $P \lhd X \Rightarrow P \mid\equiv (P \lhd X)$,

A6. $\rightarrow^K P \Rightarrow P \mid\equiv \rightarrow^K P$,

A7. $P \mid\equiv \alpha \wedge P \mid\equiv \beta \Rightarrow P \mid\equiv, \alpha \wedge \beta$,

A8. $P \rhd X \Rightarrow Q \lhd X$ for some $Q \neq P$,

A9. $P \rhd \{X^S\}_K \wedge \rightarrow^K P \Rightarrow P \rhd X$.

3. Model of Computation

A system consists of a finite collection of agents $P_1, \ldots, P_n$ who comunicate by sending messages to each other. We assume the existence of a distinguished agent P_e called *environment* which plays a role of other principals trying to attack an authentication protocol.

At any given time an agent P_i (for $i = 1, \ldots n, e$) is in some *local state* (s_i), *a global state* is a tuple $s = (s_e, s_1, \ldots, s_n)$ of local states. In any given state any principal can change its local state by performing an action. An action is identified with a state-transition relation. *A run* is an infinite sequence of global states $r = (s^0, s^1, \ldots)$. By *a system* $\mathfrak{R}$ we mean a set of runs. Integer times are assigned to each global state in a run: the first state of a run r, called *the initial state,* is assigned by the time 0, and the k th state is assigned by the time $k-1$. We denote the global state at time k in a run r by an ordered pair (r,k).

We assume a principal P_i's local state includes *a local history* h_m^k - the sequence of all actions the principal has performed up to the time k and *a set of encryption keys* K_i^k the principal holds in the time k. The environment P_e's state includes a *global history* - the sequence of actions any principal has performed, a key set K_e^k and *a message buffer* m_i for each system principal P_i containing all messages sent to P_i but not yet delivered.

We assume that the set of actions a principal P_m can perform includes the following actions:

1. $(P_m {}^{\rightarrow}Q; M)$ – P_m sends the message M to Q, the message M is added to Q's message buffer.

2. $({}^{\rightarrow}P_m; M)$ – P_m obtains the message M, M is nondeterministically chosen and deleted from P_m message buffer.

3. $K^{\rightarrow}P_m$ - the key K is added to K_m.

4. Semantics

Now we give our semantics for the logic. We define the truth of $\alpha \in F$ in point (r,k) of the system $\mathfrak{R}$ (denote $(r,k) \models \alpha$) by induction on the structure of α :

$(r,k) \models P_m \rhd X$ iff $({}^{\rightarrow}P_m; X)$ is an action occuring in h_m^k,

$(r,k) \models P_m \lhd X$ iff $(P_m^{\rightarrow}Q; X)$ is an action of h_m^k, for some $Q \in T_P$,

$(r,k) \models \rightarrow^K P_m$ iff $K \in K_m^k$,

$(r,k) \models \neg\alpha$ iff $(r,k) \models \alpha$ does not hold,

$(r,k) \models \alpha \wedge \beta$ iff $(r,k) \models \alpha$ and $(r,k) \models \beta$,

$(r,k) \models \alpha \vee \beta$ iff $(r,k) \models \alpha$ or $(r,k) \models \beta$,

$(r,k) \models \alpha \Rightarrow \beta$ iff $(r,k) \models \neg\alpha$ or $(r,k) \models \beta$,

$(r,k) \models \alpha \equiv \beta$ iff $(r,k) \models \alpha \Rightarrow \beta$ and $(r,k) \models \beta \Rightarrow \alpha$, We define the truth of $(r,k) \models P_i|\equiv \alpha$ (in the point (r,k) of system) by induction on α :

$$(r,k) \models P_i|\equiv (Pi \rhd X) \quad \text{iff} \quad (r,k) \models P_i \rhd X,$$

$$(r,k) \models P_i|\equiv (Pi \lhd X) \quad \text{iff} \quad (r,k) \models P_i \lhd X,$$

$$(r,k) \models P_i|\equiv \rightarrow^K P_m \quad \text{iff} \quad (r,k) \models \rightarrow^K P_m,$$

for another formulas:

$$(r,k) \models P_i|\equiv \alpha \quad \text{iff} \quad \alpha \in A_0 \text{ or for some formula } \beta$$
$$(r,k) \models P_i|\equiv \beta \text{ and } (r,k) \models P_i|\equiv (\beta \Rightarrow \alpha).$$

A formula α is *a tautology* if $(r,k) \models \alpha$, for any (r,k).

A formula α is *a semantic consequence* of the set of formulas X (denoted: $X \models \alpha$) iff for any point of the system (r,k) and for any formula β from the set X if $(r,k) \models \beta$, then $(r,k) \models \alpha$.

A point (r,k) is called *a model* for a set of formulas X iff for any formula α from $X, (r,k) \models \alpha$.

5. Completeness

Now we present some important properties of our logic.

Let $S = (R, A)$ be a logical system ($R = \{r_o\}$, A is the set of axioms: $A_0 \cup \{A1, \ldots, A9\}$. Let F_a be the set of all formulas in a form: $P \lhd X, P \rhd X, \rightarrow^K P$, for any P, X i K.

Lemma 1.

If $\alpha \notin X$ $[X = Cn(R, A \cup X)]$, where X is a set of formulas, such that $X \cap F_a$ is a finite set, then exists a set of formulas Y such that:

1. $\alpha \notin Y (Y \neq F)$,
2. $X \subseteq Y$,
3. $Cn(R,Y) = Y$,
4. $\forall_{\beta \in F} (\beta \in Y \vee \neg\beta \in Y)$,
5. $\forall_{\beta \notin Y} (\alpha \in Cn(R, Y \cup \{\beta\}))$.
6. $Y \cap F_a$- is a finite set.

Proof. (sketch)
Let $F = (\varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_n, \ldots)$ be a sequence of all formulas of our language.
Let $(X_n)_{n \in N}$ be a sequence of sets such that:
$X_0 = X$,
$X_{n+1} = Cn(R, X_n \cup \{\varphi_n\})$ when:

1. $\alpha \notin Cn(R, A \cup X_n \cup \{\varphi_n\})$,
2. $\varphi_n \in F_a$,
3. $\varphi_n \in Cn(R, A \cup X_n)$.

$Cn(R, X_n \cup \{\neg\varphi_n\})$ otherwise.

Let $Y = \bigcup\limits_{n \in N} X_n$.

Lemma 2.

If X is a set of formulas such that:

1. $X = Cn(R, A \cup X)$,
2. $X \cap F_a$ - is a finite set,

then X has a model.

Theorem (completeness)
If X is a set of formulas such that $X \cap F_a$ is a finite set, then:

$$\alpha \in Cn(R, A \cup X) \quad \text{iff} \quad X \models \alpha$$

Proof: (sketch)
,,⇒"
We can show that the axioms are tautologies and derivations preserve truth.
,,⇐"
We assume: $\alpha \notin Cn(R, A \cup X)$. By Lemma 1 there exists a set Y such that $\neg\alpha \in Y$. By Lemmas 1 and 2, Y has a model. I.e.: $(r, k) \models \neg\alpha$, for some (r, k). Since $X \subseteq Y$, (r, k) is a model of X too. Thus α is not a semantical consequence of X.

6. Conclusions

In this paper we have presented a new complete logic of authentication. We have introduced reformulated language, axiom inference system, model of computation and semantics. We hope that this logic will be usefull in the research of properties of cryptographic protocols. The most important of these are: determining the identity of agents who communicate to each other and possibility of decryptioning a ciphertext by intruder. To check wheather the protocol has got a property we transform the protocol and the property into the special logical formulas, properly α and β, and we check is the formula $\alpha \Rightarrow \beta$ is a tautology in our logic or not. Testing the properties in the known cryptographic protocols will be a field for the future work.

7. References

[NS78] Needham R., Schroeder M., Using encryption for authentication in large networks of computers, Communications of the ACM, 21(12), (1978) pp. 993-999.

[BAN89] Burrows M., Abadi M. and Needham R. A logic of authentication. Research Report 39, Digital Systems Research Center, February 1989.

[GNY90] Gong L., Needham R., Yahalom R., Reasoning about belief in cryptographic protocols, In Proceedings of the 1990 IEEE Symposium on Security and Privacy, (1990) pp. 234-248, IEEE Computer Society Press.

[AT91] Abadi M. and Tuttle M. A semantics for a logic of authentication. In Proceedings of the tenth ACM Symposium on Principies of Distributed Computing, pp. 201 - 216, ACM Press, August 1991.

[S91] Syverson P., The use of logic in the analysis of cryptographic protocols, In Proceedings of the 1991 IEEE Computer Security Symposium on Security and Privacy, (1991) pp. 156-170, IEEE Computer Society Press.

[SM93] Syverson P., Meadows C., A logical language for specifying cryptographic protocol requirements, In Proceedings of the 1993 IEEE Computer Security Symposium on Security and Privacy, (1993) pp. 165-177, IEEE Computer Society Press.

[SvO94] Syverson P., van Oorschot P.C., On unifying some cryptographic protocol logics, In Proceedings of the 1994 IEEE Computer Security Foundations Workshop VII, (1994) pp. 14-29, IEEE Computer Society Press.

Institute of Mathematics and Computer Science
Pedagogical University of Częstochowa
e-mail: m.kurkowski@wsp.czest.pl