Probabilistic Authentication Analysis

Josep Domingo-Ferrer
Departament d'Enginyeria Química
Escola Tècnica Superior d'Enginyeria
Universitat Rovira i Virgili
Autovia de Salou, s./n.
E-43006 Tarragona, Catalonia-Spain
Tel. +34-77-559657
Fax +34-77-559710
e-mail jdomingo@etse.urv.es

Abstract

This paper presents an application of probabilistic reasoning to the analysis and evaluation of authentication protocols in a scenario with uncertainty. The starting point is Burrows-Abadi-Needham (BAN) logic, which is an invaluable tool for modelling, designing and debugging authentication protocols without uncertainty. Denoting by "principal" any of the parties involved in a protocol —smart cards, teller machines, workstations, file servers, etc.—, a probabilistic extension of BAN logic is introduced that accomodates authentication scenarios where no principal trusts each other completely on any statement. We also propose a measure of efficiency to compare protocol behaviour in front of uncertainty. Finally, a generalization is provided for the case in which only interval estimates of certainties on assumptions are known.

1 Introduction

An authentication protocol consists of a sequence of messages exchanged by two or more parties, called principals, with the aim of elucidating the truthfulness of some given statements. For instance, a principal may wish to make sure of another principal's identity: such is the case when a bank teller requires a smart card to prove its identity. Unless communication channels between principals can be considered secure, messages exchanged during an authentication protocol are usually encrypted, in order to prevent intrusion or impersonation.

Authentication logics such as BAN [Burr90] and GNY [Gong91] are used for abstracting and modelling authentication protocols, so that their possible design flaws become apparent. Since their appearance, such logics have paid little attention to the concept of uncertainty or mistrust. Using

these tools to analyze an authentication protocol between a set of principals yields an "all-or-nothing" result: either the protocol succeeds in getting all principals convinced or else it fails and must be redesigned. However, the concept of "probabilistic guarantee" is already mentioned in the seminal paper [Burr90] as a possible improvement to BAN logic. Even if several refinements and/or criticisms have been made to BAN logic (see [Ness90], [Boyd94], [Haus92]), only [Camp92] presents an *ad-hoc* technique for dealing with partial belief. The proposal in this paper has the advantage of being more closely based on [Burr90] and using the framework of probabilistic logic.

There are situations in which the "all-or-nothing" approach to authentication analysis is rather awkward. Rather than aiming at an absolute certainty, we could more generally think of a belief being true with some probability; this leads to a probabilistic degree of certainty or trust. Authentication protocols may have to be used in scenarios where, in general, no principal trusts each other completely on anything: for example, in an open environment where smart cards from several issuers interact with third-party servers. A mistrusting principal's initial assumptions about mistrusted principals can be stated as probabilistic beliefs: for example, a principal believes that there is only some probability that another principal's public key is K.

A protocol that takes probabilistic assumptions as its starting point cannot be expected to provide absolute certainty. Instead, each principal A_i will reach at the end of the protocol a degree of certainty P_{ij} about statement X_j, with $0 \leq P_{ij} \leq 1$: this means that A_i believes that the probability of X_j being true is P_{ij}. In this way, a protocol is good if all principals are satisfied with the degrees of certainty that they obtain. $1 - P_{ij}$ will be called the degree of uncertainty or mistrust of A_i with relation to statement X_j.

In section 2, we show how BAN-like logics can be extended in order to make them able to express uncertainty as explained above. In section 3, the efficiency of a protocol in a scenario with uncertainty is characterized and measured. A complete example is given in section 4. Finally, section 5 contains a generalization to handle certainties on initial assumptions specified as interval estimates, as well as an identification of uncertainty sources other than initial assumptions.

2 Probabilistic BAN logic

In this section, BAN logic is modified according to the principles of probabilistic logic (see [Paas88] for a survey). We will use the predicate **believes**(A, X, P) to denote that principal A believes that the probability of statement X being true is P. So A views statement X as a boolean random variable having a Bernoulli distribution with parameter P: according to A, X takes the value "true" with probability P and the value "false" with probability $1 - P$.

Using a Prolog-like notation, BAN inference rules [Burr90][Abad90] can be reformulated as follows in terms of probabilistic beliefs —for the reader unfamiliar with BAN logic, *dropping probabilities in what follows yields the original BAN inference rules*—. The conclusion on the left-hand side of

each rule can be inferred if the conjunction of the premises on the right-hand side holds with nonzero probability.

- *Jurisdiction rule.*

$$\begin{aligned}&\mathbf{believes}(A, X, P_1P_2P_3) :- \\ &\quad \mathbf{believes}(A, \mathbf{controls}(B, X), P_1), \\ &\quad \mathbf{believes}(A, \mathbf{believes}(B, X, P_3), P_2).\end{aligned} \tag{1}$$

The function $\mathbf{controls}(B, X)$ expresses the "authority principle" of B over statement X: if A thinks that B believes X and that B is authoritative on X, then, by the authority principle, A also believes X. See subsection 2.3 for a justification of the product $P_1P_2P_3$ in the conclusion.

- *Public-key message-meaning rule.*

$$\begin{aligned}&\mathbf{believes}(A, \mathbf{said}(B, [X, P_2]), P_1) :- \\ &\quad \mathbf{believes}(A, \mathbf{has\_key}(B, K), P_1), \\ &\quad \mathbf{sees}(A, \mathbf{encrypted}([X, P_2], \mathbf{inv}(K))).\end{aligned} \tag{2}$$

The predicate $\mathbf{has\_key}(B, K)$ means that K is the public key of B. The function $\mathbf{encrypted}(M, K)$ denotes message M encrypted under key K; a message consists of a statement X and a probability P_2 indicating how convinced is the sender of the truth of X —*in the original BAN approach, a principal is supposed to completely believe what she encrypts at the moment of encrypting it, because such a belief can be deduced by chaining the original message-meaning and nonce-verification rules* (see below)—. The function $\mathbf{inv}(K)$ denotes the private key corresponding to public key K.

- *Shared-key message-meaning rule.*

$$\begin{aligned}&\mathbf{believes}(A, \mathbf{said}(B, [X, P_2]), P_1) :- \\ &\quad \mathbf{believes}(A, \mathbf{share\_key}(A, B, K), P_1), \\ &\quad \mathbf{sees}(A, \mathbf{encrypted}([X, P_2], K)).\end{aligned} \tag{3}$$

The predicate $\mathbf{share\_key}(A, B, K)$ means that A and B share a key which is only known to them.

- *Link message-meaning rule.*

$$\begin{aligned}&\mathbf{believes}(A, \mathbf{said\_on\_link}(B, [X, P_2], L), P_1) :- \\ &\quad \mathbf{believes}(A, \mathbf{has\_secure\_link}(B, L), P_1), \\ &\quad \mathbf{sees\_on\_link}(A, [X, P_2], L).\end{aligned} \tag{4}$$

The predicate $\mathbf{has\_secure\_link}(B, L)$ means that link L is a secure channel from B.

- *Nonce-verification rule.*

$$\begin{aligned}&\mathbf{believes}(A, \mathbf{believes}(B, X, P_3), P_1P_2) :- \\ &\quad \mathbf{believes}(A, \mathbf{fresh}(Y), P_1), \\ &\quad \mathbf{believes}(A, \mathbf{said}(B, [X, Y, P_3]), P_2).\end{aligned} \tag{5}$$

  The function $\mathbf{fresh}(Y)$ means that the nonce Y is fresh.

- *Timeliness verification rule.*

$$\begin{aligned}&\mathbf{believes}(A, \mathbf{believes}(B, X, P_3), P_1P_2) :- \\ &\quad \mathbf{believes}(A, \mathbf{timely}(L), P_1), \\ &\quad \mathbf{said}(A, \mathbf{said\_on\_link}(B, [X, P_3], L), P_2).\end{aligned} \tag{6}$$

  The function $\mathbf{timely}(L)$ means that messages sent over link L are timely delivered by the link.

2.1 On independence of beliefs

For the jurisdiction, nonce-verification and timeliness verification rules, we assume that the two beliefs on the right-hand side are statistically independent —*i. e.*, A sees the statements contained in both beliefs as Bernoulli boolean random variables independent from each other—; as to the second premise of the jurisdiction rule, B's nested belief is assumed to be independent from A's belief (refer to subsection 2.3). If assuming independence were not reasonable, then assumptions on conditional probabilities/certainties would be necessary (see [Paas88]), because a rule infers a conclusion from conjunction of two premises and, as justified in subsection 2.3, assigns to the statement in the conclusion the joint distribution of the two statements in the premises. Under the independence assumption, when a conclusion is inferred from the conjunction of two premises containing independent statements X, Y that follow Bernoulli distributions with parameters P_X and P_Y, respectively, the statement in the conclusion is assigned a Bernoulli distribution with parameter P_XP_Y (remember that the parameter represents the probability of the statement being true). In what remains, we mean by "probability of a belief" the Bernoulli parameter of the statement in the belief.

2.2 Seeing as strongly believing to have seen

The predicates $\mathbf{sees}(A, X)$ and $\mathbf{sees\_on\_link}(A, X, L)$ express facts, but they can also be thought of as being beliefs with probability 1 (A believes that the probability of her seeing X being true is 1). In this way, the following equivalences can be formulated

$$\mathbf{sees}(A, X) \Leftrightarrow \mathbf{believes}(A, \mathbf{sees}(A, X), 1) \tag{7}$$

$$\mathbf{sees\_on\_link}(A, X, L) \Leftrightarrow \mathbf{believes}(A, \mathbf{sees\_on\_link}(A, X, L), 1) \tag{8}$$

So in the public-key message-meaning rule, we are assuming that the statements has_key(B,K) and sees(encrypted(M,inv(K))) are viewed by A as independent Bernoulli random variables. An analogous assumption is made for the other message-meaning rules.

2.3 On probability computation

From subsections 2.1 and 2.2, it follows that the product is the natural function for computing the probability of the conclusion inferred from the conjunction of premises in all the above rules, except perhaps for the jurisdiction rule, where the second premise in the conjunction contains a nested belief.

If a rule right-hand side contains as premises two beliefs in statements X and Y with probabilities P_X and P_Y, then the probability associated to the belief on the left-hand side —conclusion— is actually $\geq P_X P_Y$, since the same conclusion can possibly be reached from other premises. However, following a worst-case approach, we have taken the lower bound represented by the product in the previous rules.

In the case of the jurisdiction rule, the probability of the conclusion could more generally be written as $P_1 T(P_2, P_3)$, where $T(P_2, P_3)$ results from by "unnesting" P_3 from

$$\mathbf{believes}(A, \mathbf{believes}(B, X, P_3), P_2) \tag{9}$$

so that

$$\mathbf{believes}(A, \mathbf{believes}(B, X), T(P_2, P_3))$$

holds. It is reasonable to require that the function $T : [0,1] \times [0,1] \rightarrow [0,1]$ satisfy the following properties $\forall P, Q, R, S$

$$\begin{aligned} T(P,1) &= P \\ T(P,Q) &\leq T(R,S) \quad \text{if } P \leq R \text{ and } Q \leq S \\ T(P,Q) &= T(Q,P) \\ T(P,T(Q,R)) &= T(T(P,Q),R) \end{aligned} \tag{10}$$

The associativity property can be checked for consistency on a doubly nested extension of equation 9

$$\mathbf{believes}(A, \mathbf{believes}(B, \mathbf{believes}(C, X, P_3), P_2), P_3) \tag{11}$$

using the independence of the principals' beliefs. Functions satisfying the properties 10 are called triangular t-norms ([Schw83]) and any of them can be taken as T; the bivariant minimum and product functions are examples of triangular t-norms. In the jurisdiction rule above, the product $T(P_2, P_3) = P_2 P_3$ has been chosen for uniformity with the products originating from conjunctions.

2.4 On assigning certainties to assumptions

Any assumption made about the initial state of a protocol can be expressed as a probabilistic belief including the believer's *subjective* probability of the statement in that particular assumption being true. If a principal is too uncertain about some statement to supply a single number, then an interval estimate for that subjective probability can be specified instead (see subsection 5.1).

3 Protocol matrices and efficiency

When starting protocol analysis, the initial state can be formalized by a vector of probabilistic assumptions or beliefs by each principal. On the other hand, each principal can have a vector of goal beliefs, whose initial probability is assumed to be zero and is supposed to increase as the protocol proceeds. The set of initial assumptions is assumed to be minimal, in that an assumption has nonzero probability only if it is needed to reach the goal beliefs. Let $P^{(0)}$ be the *initial certainty matrix*, having a row for each principal A_i and a column for each distinct statement X_j in the initial assumptions or the goal beliefs. The components of $P^{(0)}$ are zero except those corresponding to principals' certainties in their initial assumptions.

After analyzing an authentication protocol *Auth* using the probabilistic BAN inference rules described in the previous section, we obtain a *final certainty matrix* $Auth(P^{(0)})$ differing from $P^{(0)}$ only in the certainties in the principals' goal beliefs, which have the values resulting from the logic.

Definition 1 (Protocol efficiency) *Given a protocol Auth, an initial certainty matrix $P^{(0)}$ and a set G of goal beliefs, the protocol efficiency $\rho(Aut, P^{(0)}, G)$ can be defined as the average probability of the goal beliefs reached by the principals at the end of the protocol (average of the components of $Auth(P^{(0)})$ corresponding to beliefs in G), divided by the average probability of the initial assumptions (average of the nonzero components of $P^{(0)}$).*

The larger $\rho(Aut, P^{(0)}, G)$, the more efficient is the protocol. Given a set of goal beliefs and a set of assumptions, efficiency is a numerical criterion for comparing protocols.

4 An example: the Otway-Rees protocol under probabilistic assumptions

In this section, we will show how the previous concepts can be applied to a practical protocol, such as the Otway-Rees shared-key authentication protocol [Otwa87], which is also analyzed in the original BAN article [Burr90].

A description of the protocol is given below, with a and b being two principals —*e. g.*, smart-cards— having private keys k_{as} and k_{bs}, and s being the authentication server. a and b generate the nonces n_a, n_b and m; s generates k_{ab}, which becomes the session key between a and b.

1. $a \rightarrow b$: $m, a, b, \mathbf{encrypted}([n_a, m, a, b], k_{as})$
2. $b \rightarrow s$: $m, a, b, \mathbf{encrypted}([n_a, m, a, b], k_{as}), \mathbf{encrypted}([n_b, m, a, b], k_{bs})$
3. $s \rightarrow b$: $m, \mathbf{encrypted}([n_a, k_{ab}], k_{as}), \mathbf{encrypted}([n_b, k_{ab}], k_{bs})$
4. $b \rightarrow a$: $m, \mathbf{encrypted}([n_a, k_{ab}], k_{as})$

The BAN idealization of the protocol is the following, where n_c corresponds to m, a, b in the above description

1. $a \rightarrow b$: $\mathbf{encrypted}([n_a, n_c], k_{as})$
2. $b \rightarrow s$: $\mathbf{encrypted}([n_a, n_c], k_{as}), \mathbf{encrypted}([n_b, n_c], k_{bs})$
3. $s \rightarrow b$: $\mathbf{encrypted}([n_a, \mathbf{share\_key}(a, b, k_{ab}), \mathbf{said}(b, n_c)], k_{as})$, $\mathbf{encrypted}([n_b, \mathbf{share\_key}(a, b, k_{ab}), \mathbf{said}(a, n_c)], k_{bs})$
4. $b \rightarrow a$: $\mathbf{encrypted}([n_a, \mathbf{share\_key}(a, b, k_{ab}), \mathbf{said}(b, n_c)], k_{as})$

To analyze the protocol, we give probabilities to the assumptions made in [Burr90] (see also subsection 5.1).

$$
\begin{aligned}
&\mathbf{believes}(a, \mathbf{share\_key}(a, s, k_{as}), 1).\\
&\mathbf{believes}(a, \mathbf{controls}(s, \mathbf{share\_key}(a, b, K)), .8).\\
&\mathbf{believes}(a, \mathbf{controls}(s, \mathbf{said}(b, M)), .8).\\
&\mathbf{believes}(a, \mathbf{fresh}(n_a), 1).\\
&\mathbf{believes}(a, \mathbf{fresh}(n_c), 1).\\
&\mathbf{believes}(b, \mathbf{share\_key}(b, s, k_{bs}), .9).\\
&\mathbf{believes}(b, \mathbf{controls}(s, \mathbf{share\_key}(a, b, K)), .7).\\
&\mathbf{believes}(b, \mathbf{controls}(s, \mathbf{said}(a, M)), .7).\\
&\mathbf{believes}(b, \mathbf{fresh}(n_b), 1).\\
&\mathbf{believes}(s, \mathbf{share\_key}(a, s, k_{as}), .8).\\
&\mathbf{believes}(s, \mathbf{share\_key}(a, b, k_{ab}), .7).\\
&\mathbf{believes}(s, \mathbf{share\_key}(b, s, k_{bs}), .7).
\end{aligned}
\tag{12}
$$

Due to our probabilistic approach, another set of assumptions must be made concerning the probabilistic beliefs of principals in what they encrypt during the protocol. Note that the assumption

$$\textbf{believes}(s, \textbf{share\_key}(a, b, k_{ab}), .7)).$$

should be among the following, but is already in the set 12.

$$\begin{aligned}
&\textbf{believes}(a, n_a, 1). \\
&\textbf{believes}(a, n_c, 1). \\
&\textbf{believes}(b, n_b, 1). \\
&\textbf{believes}(b, n_c, 1). \\
&\textbf{believes}(s, n_a, 1). \\
&\textbf{believes}(s, \textbf{said}(b, [n_c, 1]), .9). \\
&\textbf{believes}(s, n_b, 1). \\
&\textbf{believes}(s, \textbf{said}(a, [n_c, 1]), .9).
\end{aligned} \tag{13}$$

Notice that messages being said include a probability indicating the sayer's certainty. Similarly, the probabilities in the set of beliefs 13 should be appended to the corresponding statements inside the **encrypted** calls in the BAN idealization specified after the protocol description. According to the [Burr90] analysis, the set G of protocol goal beliefs is (after addition of certainties as unknown variables)

$$\begin{aligned}
&\textbf{believes}(a, \textbf{share_key}(a, b, k_{ab}), P_{a8}). \\
&\textbf{believes}(a, \textbf{believes}(b, n_c, 1), P_{a9}). \\
&\textbf{believes}(b, \textbf{share_key}(a, b, k_{ab}), P_{b8}). \\
&\textbf{believes}(b, \textbf{said}(a, [n_c, 1]), P_{b10}).
\end{aligned} \tag{14}$$

The statements in the assumptions and the target beliefs are

$$\begin{aligned}
X_0 &:= \textbf{share_key}(a, s, k_{as}) \\
X_1 &:= \textbf{controls}(s, \textbf{share_key}(a, b, K)) \\
X_2 &:= \textbf{controls}(s, \textbf{said}(b, M)) \\
X_3 &:= \textbf{fresh}(n_a) \\
X_4 &:= \textbf{fresh}(n_c) \\
X_5 &:= \textbf{share_key}(b, s, k_{bs}) \\
X_6 &:= \textbf{controls}(s, \textbf{said}(a, M)) \\
X_7 &:= \textbf{fresh}(n_b) \\
X_8 &:= \textbf{share_key}(a, b, k_{ab}) \\
X_9 &:= \textbf{believes}(b, n_c, P_{b9})
\end{aligned} \tag{15}$$

$$
\begin{aligned}
X_{10} &:= \mathbf{said}(a, [n_c, 1]) \\
X_{11} &:= \mathbf{said}(b, [n_c, 1]) \\
X_{12} &:= n_a \\
X_{13} &:= n_b \\
X_{14} &:= n_c
\end{aligned}
$$

So we have the following initial certainty matrix

$$
P^{(0)} = \begin{pmatrix}
1 & .8 & .8 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & .7 & 0 & 0 & 0 & .9 & .7 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
.8 & 0 & 0 & 0 & 0 & .7 & 0 & 0 & .7 & 0 & .9 & .9 & 1 & 1 & 0
\end{pmatrix}
$$

After using the probabilistic BAN logic in section 2 to prove the goal beliefs from the assumptions and the messages exchanged during the protocol, we obtain the following final certainty matrix

$$
Otway - Rees(P^{(0)}) = \begin{pmatrix}
1 & .8 & .8 & 1 & 1 & 0 & 0 & 0 & .56 & .72 & 0 & 0 & 1 & 0 & 1 \\
0 & .7 & 0 & 0 & 0 & .9 & .7 & 1 & .441 & 0 & .567 & 0 & 0 & 1 & 1 \\
.8 & 0 & 0 & 0 & 0 & .7 & 0 & 0 & .7 & 0 & .9 & .9 & 1 & 1 & 0
\end{pmatrix}
$$

In this way, the protocol efficiency is

$$
\rho(Otway - Rees, P^{(0)}, G) =
$$

$$
\frac{(.56 + .72 + .441 + .567)/4}{(1 + .8 + .8 + 1 + 1 + 1 + 1 + .7 + .9 + .7 + 1 + 1 + 1 + .8 + .7 + .7 + .9 + .9 + 1 + 1)/20}
$$

$$
= 0.639
$$

5 Generalization

We have shown how to adapt BAN logic for analyzing authentication protocols starting from probabilistic assumptions in a scenario with generalized uncertainty. For this kind of scenarios, a characterization and a measure of protocol efficiency have been proposed. The basic idea can be generalized in several ways, some of which are sketched below.

5.1 Interval estimates for initial certainties

Probabilistic logic provides a first generalization concerning the initial assumptions. If there is some difficulty in determining exactly the subjective certainties in the initial assumptions, then interval

estimates of them can be used instead (see [Paas88]). Under the hypothesis of independence, inference rules multiply interval estimates of certainties as follows (rather than multiplying point certainties)

$$\prod_{i=1}^{n}[P_i, P_i']_{\alpha_i} = [\prod_{i=1}^{n} P_i, \prod_{i=1}^{n} P_i']_{\max_{1\leq i\leq n}\alpha_i}$$

where $P_i \leq P_i'$ and $0 \leq P_i, P_i' \leq 1$ for $i = 1, \cdots, n$, and where α_i is the significance level for interval $[P_i, P_i']$ (typically $\alpha_i = 0.01$ or $\alpha_i = 0.05$). The components of the final certainty matrix that is obtained are interval estimates.

Example. The interval jurisdiction rule becomes

$$\begin{aligned}&\textbf{believes}(A, X, [P_1P_2P_3, P_1'P_2'P_3']_{\max(\alpha_1,\alpha_2,\alpha 3)}) : - \\ &\quad \textbf{believes}(A, \textbf{controls}(B, X), [P_1, P_1']_{\alpha_1}), \\ &\quad \textbf{believes}(A, \textbf{believes}(B, X, [P_3, P_3']_{\alpha_3}), [P_2, P_2']_{\alpha_2}).\end{aligned} \tag{16}$$

□

It is necessary to modify the efficiency measure, so that it takes into account the interval centers, widths and significance levels: a good interval estimate for a probability is narrow, with a significance level close to 0 and a center close to 1. A possibility would be to transform intervals $[P_i, P_i']_{\alpha_i}$ into point values P_i'' defined as

$$P_i'' = (1 - \alpha_i)\frac{(P_i + P_i')/2}{1 + (P_i' - P_i)}$$

and use the efficiency measure of section 3 on the P_i'' values.

5.2 Other sources of uncertainty

BAN probabilistic logic can be extended to consider sources of uncertainty other than the initial assumptions. For instance,

- Zero-knowledge proofs (or ZKPs, see [Gold89]). As pointed out by [Haus92], BAN-like logics do not model this kind of proofs. A ZKP consists of a sequence of rounds between a prover and a verifier, so that, at the end of the j-th round, the verifier knows that what the prover wants to prove is true with probability $\geq 1 - 2^{-j}$; at the same time, it holds that the only knowledge gained during the proof by the verifier about what is to be proven is its probability of being true. ZKPs are already being used in smart-card protocols and constitute another source of uncertainty.
- Weak encryption algorithms. Speed increase or cost reduction may lead the protocol designer to using encryption algorithms that have a non-discardable probability of being broken. This is also a source of uncertainty.

Acknowledgment

Special thanks go to Prof. V. Torra for his suggestion about t-norms.

References

[Abad90] M. Abadi, M. Burrows, C. Kaufman and B. Lampson, *Authentication and Delegation with Smart-Cards*, DEC SRC Report no. 67, Oct. 1990.

[Boyd94] C. Boyd and W. Mao, "On a limitation of BAN logic", in *Advances in Cryptology - EUROCRYPT'93*, ed. T. Helleseth, Springer-Verlag, 1994, pp. 240-247.

[Burr90] M. Burrows, M. Abadi and R. Needham, *A Logic of Authentication*, DEC SRC Report no. 39, Feb. 1989 (revised Feb. 1990).

[Camp92] E. A. Campbell, "Partial belief and probabilistic reasoning in analysis of secure protocols", in *Proceedings from the Computer Security Foundations Workshop V*, IEEE, 1992.

[Gold89] S. Goldwasser, S. Micali and C. Rackoff, "The knowledge complexity of interactive proof systems", *SIAM Journal of Computing*, vol. 18, no. 1, 1989, pp. 186-208.

[Gong91] L. Gong, R. Needham and R. Yahalom, "Reasoning about belief in cryptographic protocols", in *Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy*, 1991, pp. 234-238.

[Haus92] R. C. Hauser and E. S. Lee, "Verification and modelling of authentication protocols", in *Computer Security - ESORICS 92*, eds. Y. Deswarte, G. Eizenberg and J.-J. Quisquater, Springer-Verlag, 1992, pp. 141-154.

[Ness90] D. M. Nessett, "A critique of the Burrows, Abadi and Needham logic", *ACM Operating Systems Review*, vol. 24, no. 2, 1990, pp. 35-38.

[Otwa87] D. Otway and O. Rees, "Efficient and timely mutual authentication", *ACM Operating Systems Review*, vol. 21, no. 1, 1987, pp. 8-10.

[Paas88] G. Paass, "Probabilistic logic", in *Non-Standard Logics for Automated Reasoning*, eds. P. Smets, E. H. Mamdani, D. Dubois and H. Prade, Academic Press, 1988, pp. 213-251.

[Schw83] B. Schweizer and A. Sklar, *Probability metric spaces*, North-Holland, 1983.