Bilateral Key Exchange analysed in BAN logic

Martijn Warnier

Computing Science Institute, University of Nijmegen
Toernooiveld 1, 6525 ED Nijmegen, The Netherlands
{warnier}@cs.kun.nl

February 2002

Abstract We look at an analysis of the Bilateral Key Exchange with Public Key Protocol in BAN logic. This reveals some problems with both the protocol as well as BAN logic.

1 The protocol

The *Bilateral Key Exchange with Public Key Protocol* (protocol 6.6.6. from [2]) is a simple protocol (just 3 lines) for distributing a symmetric key for use between two principals **A** and **B**. The usual notation for such a protocol can be seen in figure 1:

1. $\mathbf{B} \rightarrow \mathbf{A} : B, \{N_b, B\}_{pk_a}$
2. $\mathbf{A} \rightarrow \mathbf{B} : \{Sha(N_b), N_a, A, K_{ab}\}_{pk_b}$
3. $\mathbf{B} \rightarrow \mathbf{A} : \{Sha(N_a)\}_{K_{ab}}$

Figure 1. Bilateral Key Exchange with Public Key Protocol

Where:
pk_a is the public key of **A**.
pk_b is the public key of **B**.
K_{ab} is the session key to be exchanged (generated by **A**).
Sha is a Hash function, i.e. a function from domain to codomain in such a way that the inverse function from codomain to domain is hard to find.
N_a, N_b are nonces (randomly) generated by **A** and **B** respectively.
A, B are plain text messages containing **A**'s and **B**'s identity respectively.

Informally what happens is: (1) **B** encrypts with **A**'s public key a nonce N_b and a plain text message B, together with the same *unencrypted* plain text message B, these are sent to **A**. (2) **A** receives the message, decrypts it with its private Key pk_a^{-1} and sends a response to **B** containing a Hash from **B**'s nonce N_b, a new nonce N_a generated by **A**, the plain text message A and the session key K_{ab}. All are encrypted under **B**'s public key pk_b. (3) **B** receives and decrypts the message and sends a response to **A** containing a Hash from **A**'s nonce N_a encrypted using the session key K_{ab}.

2 BAN logic

BAN logic [1] is a logic developed for reasoning about protocols as the one described above. It is a logic about authentication and it is assumed that principals are honest.

2.1 Notation

Figure 2 shows the main symbols used in BAN logic and their (informal) meaning.

$P \mid\equiv X$	P believes X	$P \xleftrightarrow{K} Q$	K is a (good) symmetric key for P and Q
$P \mid\sim X$	P once said X	$\{X\}_K$	X is encrypted with key K
$P \triangleleft X$	P sees X	$\xmapsto{pk_p} P$	P has pk_p
pk_P	Public key of principal P	$P \mid\Rightarrow X$	P has jurisdiction over X
pk_P^{-1}	Private key of principal P	$\sharp(X)$	X is fresh

Figure 2. BAN symbols

2.2 Safe protocols

Typically one wants to prove that a certain protocol is *safe*. Which involves proving that a key is *good*. In BAN logic this means proving the following:

1. $A \mid\equiv A \xleftrightarrow{K_{ab}} B$
2. $B \mid\equiv A \xleftrightarrow{K_{ab}} B$
3. $A \mid\equiv B \mid\equiv A \xleftrightarrow{K_{ab}} B$
4. $B \mid\equiv A \mid\equiv A \xleftrightarrow{K_{ab}} B$

Analysing a protocol in BAN logic typically involves three steps: (i) idealize the protocol, (ii) find suitable assumptions and (iii) apply the natural induction rules from the BAN logic.

2.3 Idealized protocol

Since the notation in the literature for protocols is not suitable for formal analysis we need to *idealize* our protocol (see Figure 1). A message in idealized form is a formula in BAN logic. This means that the information in the protocol must be coded in BAN formulas. The process of idealizing a protocol is not formally defined, but this step is crucial in an analysis. Wrong idealizations lead to incomplete (our worse: totally wrong) proofs. Our protocol in idealized form can be seen in Figure 3.

$$\begin{aligned}
\text{Message 1}: &\quad B \rightarrow A : \{B \mid\sim N_b\}_{pk_a} \\
\text{Message 2}: &\quad A \rightarrow B : \{A \mid\sim N_b, N_a, A \xleftrightarrow{K_{ab}} B\}_{pk_b} \\
\text{Message 3}: &\quad B \rightarrow A : \{N_a, A \xleftrightarrow{K_{ab}} B\}_{K_{ab}}
\end{aligned}$$

Figure 3. Idealized protocol, suitable for analysis in BAN logic

In the next section we look at the assumptions we need.

2.4 Assumptions

What we think is a suitable set of assumptions is listed in Figure 4.

Assumption 1 and 2 assert that both A and B believe that their nonces are *fresh* (i.e. not used before invoking the protocol). Assumption 3 and 4 say that B believes A is capable of generating a good session key K_{ab} and that A believes this as well. Assumptions 5 to 8 concern the public

1. $A \mid\equiv \sharp(N_a)$
2. $B \mid\equiv \sharp(N_b)$
3. $B \mid\equiv (A \mid\Rightarrow A \overset{K_{ab}}{\longleftrightarrow} B)$
4. $A \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$
5. $A \mid\equiv \overset{pk_a}{\longmapsto} A$
6. $B \mid\equiv \overset{pk_b}{\longmapsto} B$
7. $A \mid\equiv \overset{pk_b}{\longmapsto} B$
8. $B \mid\equiv \overset{pk_a}{\longmapsto} A$
9. $A \mid\equiv \overset{pk_a^{-1}}{\longmapsto} A$
10. $B \mid\equiv \overset{pk_b^{-1}}{\longmapsto} B$

Figure 4. Assumptions for the idealized protocol

keys pk_a and pk_b. Both A and B know the public key of themselves and each other. Assumption 9 and 10 say that both A and B have their own private key.

Now we can continue with applying the BAN rules and do a formal analysis of the protocol.

2.5 Analysis

The analysis of the protocol involves applying a number of rules. This set of rules is presented in figure 5. In this figure An refers to assumption n, n refers to line n in the analysis (n $\in$ N) and the rules (which are listed in appendix A) are in **bold face**.

From A4, 7, 8 and 13 we conclude that the protocol can *in principle* safely be used. Or, in other words, that the key K_{ab} is a good session key for communication between A and B.

Before message 1 is sent we know:

0. $B \mid\!\sim \{N_b\}_{pk_a}$

Sending message 1 leads to:

1. $A \triangleleft \{B \mid\!\sim N_b\}_{pk_a}$ M1, **S**
2. $A \triangleleft B \mid\!\sim N_b$ 1, A9, **PK**

Sending message 2 leads to:

3. $B \triangleleft \{N_b, N_a, A \overset{K_{ab}}{\longleftrightarrow} B\}_{pk_b}$ M2, **S**
4. $B \mid\equiv A \mid\!\sim (N_b, N_a, A \overset{K_{ab}}{\longleftrightarrow} B)$ 0, 3, A2, A9, A10, **EPK**
5. $B \mid\equiv \sharp(N_b, N_a, A \overset{K_{ab}}{\longleftrightarrow} B)$ A2, **FD**
6. $B \mid\equiv A \mid\equiv N_b, N_a, A \overset{K_{ab}}{\longleftrightarrow} B$ 4, 5, **NV**
7. $B \mid\equiv A \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$ 6, **W**
8. $B \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$ 7, A3, **J**

Sending message 3 leads to:

9. $A \triangleleft \{N_a, A \overset{K_{ab}}{\longleftrightarrow} B\}_{K_{ab}}$ M3, **S**
10. $A \mid\equiv B \mid\!\sim (N_a, A \overset{K_{ab}}{\longleftrightarrow} B)$ 10, A4, **SK**
11. $A \mid\!\sim \sharp(N_a, A \overset{K_{ab}}{\longleftrightarrow} B)$ A1, **FD**
12. $A \mid\equiv B \mid\equiv (N_a, A \overset{K_{ab}}{\longleftrightarrow} B)$ 10, 11, **NV**
13. $A \mid\equiv B \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$ 12, **W**

Figure 5. Analysis of the idealized protocol

3 Discussion

There are a number of known problems with BAN logic. First of all, the idealization step has no formal definition (as noted in section 2.3). For example, it is not at all clear that $\mathbf{B} \rightarrow \mathbf{A} : B, \{N_b, B\}_{pk_a}$ is the same as $B \rightarrow A : \{B \mid\sim N_b\}_{pk_a}$. Since everyone can, in principle, mask as if he is B and send this message to A. So the result we derived in line 2 certainly is questionable.

In fact it seems that the protocol is vulnerable to an attack, where a principal (let's say a *Spy*) can send message 1 to A and intercept the reply (message 2) to B. The *Spy* can repeat this with slightly different messages and try to break pk_a^{-1} using cryptoanalysis. This could lead to compromising the key K_{ab}. Note that this kind of attack is outside the scope of BAN logic where no cryptoanalysis is permitted.

Another problem with BAN logic is that it has no clear semantics. This keeps us from giving a soundness proof of the rules of BAN logic.[1].

Yet another problem is that of the assumptions. It is not clear which assumptions we may and may not presuppose. In the case of our little protocol this is not a problem, but with larger protocols involving a number of principals this becomes a problem indeed. Note that this problem is not specific for BAN logic.

In conclusion we can say that BAN logic is suitable in cases where we want to take a "quick look" at a protocol, but not if we want to give a complete proof of "safeness" of a protocol. Approaches which use a clear semantics seem to give better results.

References

1. M. Burrows, M. Abadi, and R. Needham. A logic of authentication. *Proc. Royal Soc.*, Series A, Volume 426:233–271, 1989.
2. J. Clark and J. Jacob. A Survey of Authentication Protocol Literature, version 1.0, 1997. available at URL http://www-users.cs.york.ac.uk/ jac/papers/drareview.ps.gz.
3. L. Gong, R. Needham, and R. Yahalom. Reasoning about Belief in Cryptographic Protocols. In *Proceedings of the 1990 IEEE Computer Society Symposium on Research in Security and Privacy*, pages 234–248. IEEE Computer Society Press, 1990.
4. Paul. F Syverson and Paul C. van Oorschot. A Unified Cryptographic Protocol Logic. Technical Report 5540-227, NRL CHACS, 1996.
5. J. Wessels. Ban-logic. Technical report, CMG PUBLIC SECTOR B.V. 2001.

A BAN rules

This section contains some of the BAN rules from [1] and some extra rules which deal with public keys. Note that only rules which are used in the actual analysis are listed here.

A.1 Original BAN rules

Symmetric Key Rule

$$\frac{P \mid\equiv Q \overset{K}{\longleftrightarrow} P \qquad P \triangleleft \{X\}_K}{P \mid\equiv Q \mid\sim X} \ \text{(SK)}$$

Freshness Distribution Rule

$$\frac{P \mid\equiv \sharp(X)}{P \mid\equiv \sharp(X, Y)} \ \text{(FD)}$$

[1] There are a number of known extensions to BAN logic in the literature that try to tackle this problem by giving a semantics for BAN or extensions of BAN, see [3,4].

Nonce Verification Rule

$$\frac{P \mid\equiv \sharp(X) \qquad P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}\ \text{(NV)}$$

Weakening Rule

$$\frac{P \mid\equiv Q \mid\equiv (X, Y)}{P \mid\equiv Q \mid\equiv X}\ \text{(W)}$$

Jurisdiction Rule

$$\frac{P \mid\equiv Q \mid\Rightarrow X \qquad P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}\ \text{(J)}$$

A.2 Extended BAN rules

Some extra rules. The first two which deal with public key encryption are from Erik de Vink[2]. The last one (in subsection A.2) is from [5].

Public Key Rule

$$\frac{P \mid\equiv \xmapsto{pk_p^{-1}} P \qquad P \triangleleft \{X\}_{pk_p}}{P \triangleleft X}\ \text{(PK)}$$

Extended Public Key Rule

$$\frac{P \mid\equiv \sharp(X) \qquad P \mid\sim \{X, Y\}_{pk_q} \qquad P \triangleleft \{X, Z\}_{pk_p}}{P \mid\equiv Q \triangleleft X, Y \qquad P \mid\equiv Q \mid\sim X, Z}\ \text{(EPK)}$$

Where the usual assumptions about public and private keys are presupposed.

Sees Rule

$$\frac{\text{Message n} : P \to Q : X}{Q \triangleleft X}\ \text{(S)}$$

[2] The rules are used in a course on the Technical University Eindhoven. The security course entitled "Bewijsbare veiligheid" (en: Provable Safety).