マルチメディア通信と分散処理　69－6
（１９９５．　３．　２）

BANロジックによる ネットワーク・アドレス情報の完全性の検証

村山 優子
広島市立大学 / 情報科学部

本論文では, コンピューター・ネットワークにおけるアドレス情報について, 不正確な情報からの脅威を解説し, 情報流の完全性の必要を説き, 信用できるアドレス情報のための登録プロトコルを紹介する. さらに, そのプロトコルをBANロジックを用いてフォーマルな形での検証を試み, このような検証において, 情報流の経路の必要性を示す.

Using BAN logic for the proof of a registration protocol

Yuko Murayama
Faculty of Information Sciences/Hiroshima City University

The current problem of the computer network environment is that information held in the information systems is not necessarily trustworthy. We look particularly at an address as an example of the network configuration information, and list the serious consequences from the incorrect addresses. The current address information flow is examined, and then improved by use of certificate. This paper trys to make use of the logic introduced by Burrows, Abadi, and Needham, notably called BAN Logic, to prove the flow of the issuing procedure of a certificate. The discovery in our application is the need for incorporating the information link attribute to the knowledge obtained by the participants of information flow.

1 Introduction

In this paper, we look at network addresses as an example of information used by computer networks, and list the problems with incorrect network addresses. We propose a registration protocol for issuing a certificate of a network address, so that the network would be protected from some of the problems. The protocol is examined in a formal manner using the logic introduced by Burrows, Abadi, and Needham [2]. Although Nessett argued that that the logic missed out the confidentiality aspect intentionally for simplifying the process [3] [1], it is no problem with our use of the logic because our primary concern is to examine information integrity.

2 The current network address information flow and its problems

What is missing in the current network system is that there is no mechanism for verifying the credibility of an address and for binding between a host and the address. In the Internet environment, this could lead to the network level threats as follows:

1. **Unauthorised tampering**
2. **Unauthorised use of the resource (resource stealing)**
3. **Doubled traffic**
4. **Denial of services by a network storm**

3 Use of a certificate and a registration protocol

We propose the use of address certificates in address resolution to prevent the problems listed in the previous section, and present a registration protocol for issuing a certificate. Address resolution is the translation of a network address into the lower layer address such as a MAC address in a LAN. A certificate includes network and lower layer address mapping, and is issued by a Certification Authority(CA) only when the host configuration is verified over the network. The idea is that only certifed addresses would be used in address resolution. This prevents a source node from sending a packet to a bogus host, because the nodes invoke address resolution to locate the server. In the following we introduce a protocol for configuration confirmation as well as registration.

Besides the Naming Authority, the following agents are involved in our registration procedure: Management System (M), Certification Authority (CA), Information System (R), system (A), host (h).

The configuration confirmation sequence is as follows:

1. A system, A, uses the network to inform the manager, M, of the completion of the configuration of network software, h, using the registration ID
2. The manager, M, gathers the information on h, including the associated registration ID, which was registered at Certificate Registration.
3. The configuration of h is verified using the previously registered information
4. Upon successful verification, the manager recognises the addition of h.
5. The lower layer and network addresses are certified and a certificate is produced.
6. The certificate is registered to the information system.
7. The certificate is given to the host as well.

The following certificate issued at 5 will be used in the address resolution operation:

$$\{ \text{ lower layer address, network address, time stamp } \}SK_{CA}$$

where a lower layer address and the associated network address is encrypted by the secret key of CA.

4 Formal analysis of the protocol

4.1 Overview

We examine our registration protocol by the formal method introduced by Burrows, Abadi, and Needham Ref.[2]. Using BAN Logic, the knowledge obtained by participants is examined at the end of each protocol message. It is particularly useful to examine whether some information has been flown as expected by a protocol designer. We are interested particularly in how an allocated address is passed by its allocator, the Naming Authority, to the host and the information systems, and how its integrity is preserved.

4.2 Introduction to notation

The followings are the notations introduced by Burrows et al.:

$A \models X$: means that A believes that X is a true information item.

$A \Rightarrow X$: means that A has a jurisdiction over X.

$\sharp(X)$: originally means that the information item, X, is generated recently.

We modify this definition into that the production of X has been validated and verified recently; i.e.X is perceived as correct at a recent time, however, it is not clear whether its generation was recent, or not.

$A \triangleleft X$: A sees X.

$A \mid\sim X$: A once said X.

We introduced a new notation as follows, which indicates the information item has been notarised by a trustful authority:

$\langle\!\langle X \rangle\!\rangle_C$: X is certified by the authority C.

In the following subsections, we also use the following abbreviations:

- *Reg* : the registration ID
- X_{anet} : the network address of h
- X_{alow} : the lower layer address of h
- X_d : the description of the h
- $X_c(Z)$: the cryptographic system attributes of Z
- T_Z : the time stamp of Z

Besides the notations of participants CA, M, R, A, and h introduced in the previous section, we denote NA as the local Naming Authority.

NA assigns the network address to a network interface, and is supposed to do it reliably — i.e. it checks on duplicate addressing. On the other hand, lower layer addresses are pre-assigned by a naming authority (possibly a vendor) outside the organisation, but it would be possible for a local user to change it. Its reliability, therefore, is unknown. Our CA is supposed to check up on duplicate lower layer addresses as well as network addresses before issuing a certificate.

4.3 The goal of analysis

As the notation which we use is originally intended for authentication, the successful result of a protocol between two principals, A and B, is that both principals know the shared secret, and each knows that other knows it as well. However, our goal is different, in that we need to ensure the address mapping information, e.g.(X_{anet}, X_{alow}), generated by the *information originator*, NA, and verified by CA, is assigned to the *information provider*, A, as is intended, and arrives at an *information system*, R, eventually.

We use the the following notations:

- *Bind(Xanet)= Xalow* means the network address, *Xanet* is assigned for the network interface with a lower layer address, *Xalow*
- *Claim(Xalow)=Xanet* means the network interface with the lower layer address *Xalow* is claiming that its associated network address is *Xanet*

Then the goals are expressed as follows:

(1) $NA \models$ *Bind*(Xanet)=Xalow and (2) $NA \models$ *Claim*(Xalow)=Xanet

(3) $CA \models NA \models$ *Claim* (*Bind*(Xanet))=Xanet and (4) $CA \models$ *Claim*(*Bind*(Xanet))=Xanet

(5) $A \models NA \models$ *Claim* (*Bind*(Xanet))=Xanet and (6) $A \models$ *Claim*(*Bind*(Xanet))=Xanet

(7) $R \models CA \models$ *Claim* (*Bind*(Xanet))=Xanet and (8) $R \models NA \models$ *Claim* (*Bind*(Xanet))=Xanet

(9) $R \models$ *Claim* (*Bind*(Xanet))=Xanet

4.4 The BAN Logic rules

The following rules are defined in BAN Logic.
The first rule is that if A sees the information item, X, encrypted with B's secret key, A believes that B once said X as follows:

$$\frac{A \models \overset{K}{\mapsto} B, A \triangleleft \{X\}_{K^{-1}}}{A \models B \mid\sim X}$$

The second rule is that if A believes X is uttered only recently and B once said X, then A believes that B has said X, recently as follows:

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

The third rule is that if A believes that B has jurisdiction over X then A trusts B on the truth of X as follows:

$$\frac{A \models B \models X, A \models B \Rightarrow X}{A \models X}$$

4.5 The assumptions

We have the following assumptions:

Assumption 1: $CA \models \sharp\,(Reg)$

Assumption 2: $CA \models NA \Rightarrow Bind(\mathrm{X}anet) = \mathrm{X}alow$

Assumption 3: The timers in all the agents of concern are synchronised, so that they believe their time stamps each other.

For $x \in \{ M, R, CA, A \}$,
$x \models (\sharp\,(T_M), \sharp\,(T_{CA}), \sharp\,(T_R), \sharp\,(T_A))$

Assumption 4.1: $\{ A, M, R \} \models CA \Rightarrow (Reg, \mathrm{X}alow, \mathrm{X}d, \mathrm{X}c)$

Assumption 4.2: $\{ A, M, R \} \models NA \Rightarrow (\mathrm{X}anet)$

Assumption 5: $\{ A, M, R \} \models CA \Rightarrow \{ Claim(Bind(\mathrm{X}anet)) = \mathrm{X}anet \}$

That is, if
$\{ A, M, R \} \models CA \mid\sim x$,
where $x \in \{ Claim(Bind(\mathrm{X}anet)) = \mathrm{X}anet \}$,
then $\{ A, M, R \} \models x$.

Assumption 6: $CA \models \{ A, M, R \} \models CA \Rightarrow \{ (Bind(\mathrm{X}anet) = \mathrm{X}alow) \}$

Assumption 7: $\{ A, M, R \} \models NA \models Claim(Bind(\mathrm{X}anet)) = \mathrm{X}anet$,

if and only if
$\{ A, M, R \} \models CA \models Claim(Bind(\mathrm{X}anet)) = \mathrm{X}anet$.

Assumption 8: i.e. M has an authority to declare the addition of an object and it indicates that the verification has done.

$\{ CA \} \models M \Rightarrow$ (Add, object ID, information)

We also assume that relevant public keys are distributed previously to the manager system M, the information system R, and the object site system A.

4.6 Protocol analysis

The protocol is summarised as follows:

Message 1: $A \rightarrow M$: [Report, h's ID, $\{ Reg, T_A \} PK_M$]

Message 2.1: $M \rightarrow CA$: [Request, h's ID, $\{\{ Reg, T_M \} SK_M \} PK_{CA}$]
Message 2.2: $CA \rightarrow M$: [Reply, h's ID, $\{\{ Reg, Bind(\mathrm{X}anet) = \mathrm{X}alow, \mathrm{X}d, \mathrm{X}c(A), T_M + 1 \} SK_{CA} \} PK_M$]

Message 3.1: $M \rightarrow A$: [Request, $\{ h, \mathrm{ID.X}a, \mathrm{ID.X}d, T_M \} SK_M$]
Message 3.2: $A \rightarrow M$: [Reply, $\{ Claim(\mathrm{XX}alow) = \mathrm{XX}anet, \mathrm{XX}d, T_M + 1 \} SK_A$]

If M perceives that $PK_A \in \mathrm{X}c(\mathrm{A})$, $\mathrm{XX}anet = \mathrm{X}anet$, $\mathrm{XX}alow = \mathrm{X}alow$, and $\mathrm{XX}d = \mathrm{X}d$, M sends the following packets, otherwise the procedure stops here.

Message 4: $M \rightarrow CA$: [{ Report, Add,h, $Claim$X$alow$= X$anet$, Xd, Xc(A), T_M}SK_M]

Message 5.1: $M \rightarrow CA$: [{ Request, h, X$anet$, T_M}SK_M]

Message 5.2: $CA \rightarrow M$: [{ Reply,h, $\langle\langle Bind X anet = Xalow \rangle\rangle_{CA}$, $T_M + 1$}$SK_C A$]

Message 6: $M \rightarrow R$: [{ Report,h, $\langle\langle$ $Bind$ (X$anet$) = X$alow$$\rangle\rangle_{CA}$, T_M}SK_M]

Message 7: $M \rightarrow A$: [{ Set,h, $\langle\langle$ $Bind$(X$anet$) = X$alow$$\rangle\rangle$ CA, Xc(group), T_M}SK_M]

Protocol analysis is as follows.

From Message 1,

$$M \triangleleft Reg$$

From Message 2.1,

$$CA \models M \mid\sim Reg$$

From Message 2.2,

$$M \models CA \mid\sim (Reg, Bind(Xanet) = Xalow, Xa, Xd, Xc(A)),$$

and

$$M \models CA \Rightarrow (Reg, Bind(Xanet) = Xalow, Xd, Xc(A))$$

hence,

$$M \models (Reg, Bind(Xanet) = Xalow, Xd, Xc(A))$$

After the receipt of Message 3.2 which is the reply to the request in Message 3.1,

$$M \models A \mid\sim (Claim(XXalow) = XXanet, XXd)$$

Then M checks locally whether XX$anet$= X$anet$, XX$alow$= X$alow$, and XXd = Xd, (and the success of decryption of the packet shows $PK_A \in$ Xc(A)) or not. If so, $Bind$(X$anet$)= X$alow$ is verified as follows:

$$M \models A \mid\sim (Claim(Xalow) = Xanet)$$

From Message 4, we deduce

$$CA \models A \mid\sim (Claim(Xalow) = Xanet)$$

On receipt of Message 5.1 of the request of certificate from M, CA remembers that it already knew the followings:

$$CA \models (Reg, Claim(Xalow) = Xanet)$$

and from Message 4,

$$CA \models A \mid\sim (Claim(Xalow) = Xanet)$$

Thus CA could issue a certificate straight away. However, here CA checks the inconsistency of addressing by looking through a list of addresses which CA has certified. NA will be informed if there is any double-allocation of an address. Moreover, after the validation there needs to be a transaction (probably off-line) between CA and NA to confirm each other that the allocated network address, X$anet$ and its associated lower layer address X$alow$ are indeed usable as follows:

$$CA \rightarrow NA: \langle\langle Claim(Xalow)=Xanet, T_{CA}\rangle\rangle\ CA \qquad (5.1.add1)$$

If NA perceived that $Bind$(X$anet$)=X$alow$, then it sends the following reply:

$$NA \rightarrow CA\text{: } \langle\!\langle\ \mathrm{T}_{CA+1}\ \rangle\!\rangle\ \mathrm{NA} \qquad \text{(5.1.add2)}$$

From (5.1.add1),

$$NA \models CA \mid\sim (Claim(Xalow) = Xanet)$$

From (5.1.add2), we can see that NA admits the followings:

$$NA \models Bind(Xanet) = Xalow \qquad (1)$$

and

$$NA \models Claim(Xalow) = Xanet \qquad (2)$$

As CA already knew that $Claim(Xalow)$=X$anet$, it deduces the following:

$$CA \models NA \models (Claim(Bind(Xanet)) = Xanet) \qquad (3)$$

CA issues a certificate for $Claim(Bind(Xanet))$=X$anet$ in Message 5.2.

From Message 5.2 as CA has sent the certificate, it shows that

$$CA \models Claim(Bind(Xanet)) = Xanet \qquad (4)$$

On issue of the certificate, CA should expire the *Reg.* Also, the timeout limit should be set in *Reg* in case that for some reason the registration would never be invoked.

The rest can be proved in the similar way.

5 The information path

We have managed to use the logic to prove the goals; however, we have found a little difficulty in this use. We have no clear way to differentiate the information which is set to the host at the Configuration Phase, from the one stored in CA. For instance, in our proof, we denoted XX_{anet} as the network address in the host, and X_{anet} as the one stored in CA. For the precise information flow examination, we may need some more explicit way to express what an information item has been coming through — information paths. We could express $M \models h \models NA \models X$, but if we express $M \models (((X)_{NA})_h)$, it will have the implications that h might have made a mistake in announcing X.

6 Conclusion

The current problem in networks is that there is no way to know how reliable the information is. We looked particularly at information registration. We have introduced our registration protocol in terms of transactions between management agents. The protocol was analysed formally by making use of the logic recently introduced by Burrows, Abadi, and Needham. We have proved how the integrity of the information, an address, is maintained.
Our attempt was to use the formal notation and logic, intended originally for proving authentication protocols, to prove the integrity of information flow. One difficulty was to differentiate two addresses originated from the same address but went through the different paths. We wish to denote information paths. Apart from that, our trial of use of the logic has shown that it is useful when one wants to express the flow of information on the semantics level.

References

[1] M. Burrows, M. Abadi, and R. Needham. Rejoinder to nessett. *ACM Operating Systems Review*, Vol. 24, No. 2, pp. 39–40, April 1990.

[2] Michael Burrows, Martin Abadi, and Roger Needham. A logic of authentication. Technical Report 39, DEC Systems Research Center, February 1989.

[3] D. M. Nessett. A critique of the burrows, abadi, and needham logic. *ACM Operating Systems Review*, Vol. 24, No. 2, pp. 35–38, April 1990.