Self-Authentication of Encrypted Channels in Service Network Graph

David Lai
University of Southern Queensland
Toowoomba, Queensland, 4350
lai@usq.edu.au

Zhongwei Zhang
University of Southern Queensland
Toowoomba, Queensland, 4350
zhongwei@usq.edu.au

Abstract

Service Network Graph (SNG) was proposed as a network service sharing infrastructure to support secure services on dynamic aggregation of heterogeneous networks. To participate in SNG, a network has to share a secret key with one member of the SNG. The shared secret key will be used to set up an encrypted channel between the network and the SNG member. It is imperative to authenticate the data sent through the encrypted channel. This paper uses the symbols and approached presented by Lampson in his paper [15] to provide a formal proof of how encryption channel authenticates itself in SNG. It forms the basis of using encrypted channels in SNG.

1 Introduction

In many occasions, Internet users are not entirely happy using only the services available on their home networks. They always look forward to use services provided by other foreign networks. It is a common experience that we can extend our capability or creativity if we can take advantage of all the services that we are aware of. The reason why we cannot use all the services offered in different administrative domains is either it is too expansive, or simply impossible, to join all related networks individually by ourselves. It would be much simpler if one network can join another network and share their services to their home users. If multiple networks are linked together for service sharing, the immediate problem is how to authenticate users of the participating networks. Due to the issues of information privacy, underlying platforms, and network resources, it is practically and technically not feasible to share all the users authentication information with all linked networks.

Despite various efforts such as the use of X.509 certificates [1], trust recommendations [4, 7, 16, 18], trust establishment [6, 17, 2, 3, 5] and Kerberos [8] none has resulted in a viable solution to the problem. Our approach is to use Service Network Graph (SNG) [14, 11, 10, 13, 12]. SNG was first proposed in 2005. SNG enables the linking of heterogenous networks in an ad hoc manner to form a Service Network Graph. Within the service network graph, home users of individual networks can share the services provided by other networks within SNG. Furthermore, Dynamic Password [9] can used to complement SNG and forms an authentication protocol suite for heterogenous aggregation of ad hoc networks.

As shown in Figure 1, in order to participate in an SNG, the authentication server AS_1 of a network is required to share a secret key with the authentication server AS_2 of another network within SNG to which the network intends to attach to. The newly joined network will use this key to set up encryption channels between AS_1 and AS_2. Privacy of communications between authentication servers are protected by encryption using the shared key.

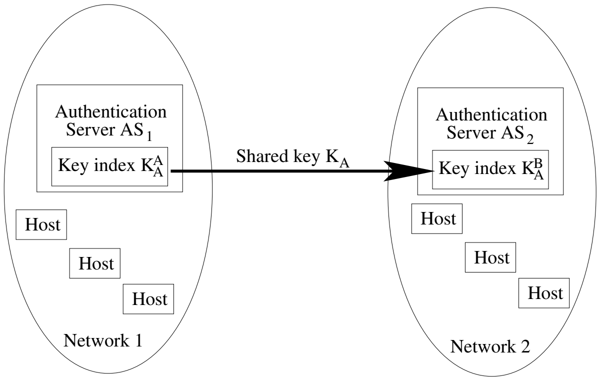

Figure 1. Network 1 joins Network 2 in an SNG

Similar to most secure transaction protocols such as SSH and Kerberos, it is important to authenticate the incoming data as well. Authenticating data helps to prevent impersonation and spoofing. SNG made use of the self-authenticating property of encrypted channels to achieve its protocol correctness. The self-authentication property can be one-way or two-way. SNG can be perceived as a net-

Original Symbol	Our Symbol	Meaning
$\Rightarrow$	$\looparrowright$	Speaks for
$\supset$	$\Rightarrow$	implies

Table 1. Symbols we use in this paper which differs from the ones used in [15]

work of networks in which the participating networks are linked together by self-authenticated encrypted channels as shown in Figure 2.

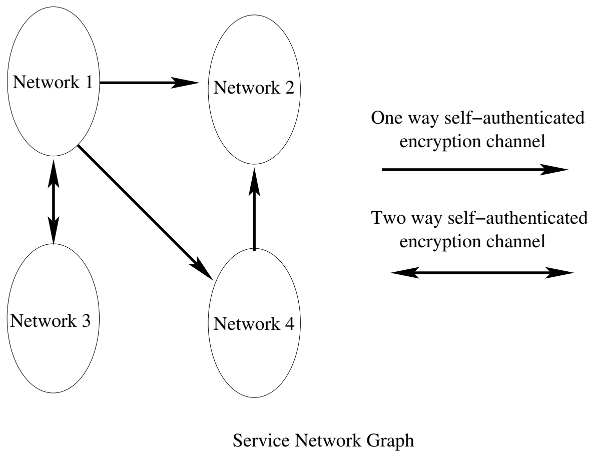
Network 1
Network 2
Network 3
Network 4
One way self–authenticated encryption channel
Two way self–authenticated encryption channel
Service Network Graph

Figure 2. Graphical representation of an SNG

In this paper we use the symbols and approach similar to that presented by Lampson [15] to give a formal proof of self-authentication for encrypted channels in SNG.

This paper is organized into four sections. In Section 2, we introduced some axioms and theorems established by Lampson in [15] first. In Section 3 we presented our proposition. We then proved the proposition by proposing and proving three lemmas. In the Conclusion section, we summarized our work in this paper and our work in the future.

2 Basic Theorems and Axioms

In this section, we listed a few basic theorems and axioms from Lampson [15] and Thayer [19]. All the terms and notations we use in this paper will be the same, unless we explicitly stated otherwise, as the ones used in Lampson's paper [15]. For instance, as listed in Table 1, we replaced $\Rightarrow$ with $\looparrowright$ to represent the "speak for" relationship; and $\supset$ with $\Rightarrow$ to represent the logical "imply" relationship.

Axiom 1 (Free encryption)
Let $\mathcal{A}$ be a set of plain text messages with elements m, m'; and $\mathcal{K}$ be a set of unique encryption keys with elements K, K'.

For $m,\ m' \in \mathcal{A}$ and $K,\ K' \in \mathcal{K}$,

$$\{m\}_K = \{m'\}_{K'} \Rightarrow m = m' \wedge K = K'$$

Free encryption means that cypher text can only be formed in one and only one way.

Next, we illustrate a basic property of the "speak for" relationship. Note that P_A, P_B and s are used to denote principal A, principal B and message s, respectively,

Theorem 1
$\vdash (P_A \looparrowright P_B) \Rightarrow ((P_A \textbf{ says } s) \Rightarrow (P_B \textbf{ says } s))$

where "$\vdash\ s$" is used to mean that "s is an axiom of the theory or is provable from the axioms". "$P_A \looparrowright P_B$" denotes "P_A speaks for P_B".

The theorem simply says if principal A speaks for principal B, then whenever A says something, B would have said the same thing.

Axiom 2 $\vdash (P_A \textbf{ says } (P_B \looparrowright P_A)) \Rightarrow (P_B \looparrowright P_A)$

This Axiom says that P_A can establish a "speak for" relationship with P_B when he declares P_B speaks for him.

Theorem 2 ***(Handoff Rule)***
$\vdash ((P_C \looparrowright P_A) \wedge (P_C \textbf{ says } (P_B \looparrowright P_A))) \Rightarrow$
$(P_B \looparrowright P_A)$

The first condition of the theorem requires $(P_C \looparrowright P_A)$. Using Theorem 1, whatever P_C says, P_A would have said the same. Hence the second condition of Theorem 2 can be rewritten as $(P_A \textbf{ says } (P_B \looparrowright P_A))$. Using Axiom 2, if P_A declares P_B speaks for him, we can arrive at the conclusion $(P_B \looparrowright P_A)$.

Theorem 1, Axiom 2, and Theorem 2 can be found in Lampson's paper [15]

3 Self-Authentication of Encrypted Channels

In this section, we will prove that an encrypted channel in SNG authenticates itself.

When a network joins an SNG as shown in Figure 1, its authentication server (AS_1) shares a unique and secret key with the authentication server (AS_2) of a member network of the SNG. The key will be stored in the key database of AS_2 and has a key index. The shared key will be used by AS_1 to establish an encrypted channel with AS_2. In so doing, AS_1 joins AS_2 and become part of the SNG.

The SNG joining process forms the basis of the assumptions of Proposition 1. The conclusion of Proposition 1 is that whatever encrypted message delivered by the encrypted channel will be originated from AS_1. In other words, encrypted channels formed when joining an SNG are self-authenticating.

The proof starts with three definitions and then three lemmas. We then apply the lemmas to prove our Proposition 1.

Definition 1 *K^r is defined as a key identifier if it allows a receiver r to know what the decryption key of an encrypted channel is but doesn't disclose anything about it to others.*

In the above definition, the key identifier can be

- an index to a database of keys kept by the receiver; or
- $\{K^{-1}\}_{K_m}$ where K_m is a secret master key kept by the receiver; or
- a pair $(K^x, \{K^{-1}\}_{K'})$, where K^x is a key identifier of key K'.

Definition 2 *When principal P_A shares a unique secret key K_A with principal P_B, key K_A and a key index K_A^B will be added to the database DB_B of keys in P_B, then we define DB_B* **says** *$(K_A^B \looparrowright P_A)$.*

It follows from the assumption that the key is unique and secret. The database in P_B is happy to say the speak for relationship assuming a one-to-one (K_A^B to P_A) mapping. Note that the mapping from P_A to K_A^B can be one-to-many.

This definition describes what happens when a unique key of one principal is shared with another principal.

Definition 3 *(Encrypted channel establishment) When P_A establishes an encrypted channel with P_B using a unique shared key K_A, then we define P_A* **says** *$(DB_B \looparrowright P_A)$.*

Note that a unique secret key has to be shared before an encrypted channel can be established.

The above definitions do not exclude the cases when a pair such as $(K_A^{B'}, \{s\}_{K_A})$ is received in place of the expected pair $(K_A^B, \{s\}_{K_A})$.

3.1 One-way Self-authentication of encrypted Channels

In this subsection, we will now show how an encrypted channel can have the property of one-way self-authentication.

Proposition 1 *(One-way Self-authentication of encrypted channels)*
Suppose:

1. *Principal P_A shares a unique secret key, K_A, with principal P_B;*
2. *P_A uses K_A to establish an encryption channel with P_B;*
3. *K_A^B is the key index of K_A^{-1} in P_B.*

then when P_B receives $(K_A^B, \{s\}_K)$, P_B can infer that P_A **says** *s.*

We will prove this by working through a sequence of lemmas.

Lemma 1 *Suppose there is only one unique decryption key corresponding to each encryption key used in an encrypted channel. Holder of a decryption key K^{-1} of an encrypted channel can infer that "K^{-1}* **says** *s" on receiving $\{s\}_K$.*

$$\vdash (K^{-1}, \{s\}_K) \Rightarrow (K^{-1}\ \textbf{says}\ s)$$

PROOF: It follows from the assumption in Lemma 1 and Axiom 1 that there is only one unique key which can get back s. We can safely associate the message s with the decryption key (index). ■

Lemma 2 *Suppose K^r is a key identifier. When a pair $(K^r, \{s\}_K)$ was received, the receiver can infer that "K^r* **says** *s".*

$$\vdash (K^r, \{s\}_K) \Rightarrow (K^r\ \textbf{says}\ s)$$

PROOF: It is a direct application of Definition 1 to Lemma 1. ■

Lemma 3 *Suppose:*

1. *Principal P_A shares a unique key K_A with principal P_B;*
2. *P_A uses K_A to establish an encryption channel with P_B;*
3. *K_A^B is the key index of K_A^{-1} in P_B.*

For P_B, $(K_A^B \looparrowright A)$.

PROOF: According to assumption 1, and Definition 2, DB_B **says** $(K_A^B \looparrowright P_A)$.

Assumption 2 and Definition 3 tells us that P_A **says** $DB_B \looparrowright P_A$ which is the same as $DB_B \looparrowright P_A$ by Axiom 2.

Since both conditions listed above are assumed to be true, we can link them into a compound predicate $(DB \looparrowright P_A) \wedge (DB\ \textbf{says}\ (K_A^B \looparrowright P_A))$. This is the predicate in the Handoff Rule from Lampson (Theorem 2): $\vdash (DB \looparrowright P_A) \wedge (DB\ \textbf{says}\ (K_A^B \looparrowright P_A)) \Rightarrow (K_A^B \looparrowright P_A)$

We may now conclude that $(K_A^B \looparrowright P_A)$ ■

PROOF of Proposition 1

The assumptions in Proposition 1 gives $(K_A^B \looparrowright P_A)$ using Lemma 3.

When P_B receives $(K_A^B, \{s\}_K)$, P_B can infer that $(K_A^B$ **says** $s)$ using Lemma 2.

According to theorem 1, we have

$\vdash (K_A^B \looparrowright P_A) \Rightarrow ((K_A^B \textbf{ says } s) \Rightarrow (P_A \textbf{ says } s))$
And hence when P_B receives $(K_A^B, \{s\}_K)$, P_B can infer that P_A **says** s. ■

3.2 Two-Way Self-authentication of encrypted Channels

Proposition 1 shows that traffic delivered to AS_2 via the encrypted channel established with the key shared by AS_1 originates from AS_1. This is a one-way self-authentication. It would be desirable if we can extend Proposition 1 to cover a two-way self-authentication. A two-way self-authentication means that message to AS_2 passing through the encrypted channel established with a key between AS_1 and AS_2 can be assured to be coming from AS_1 and those heading to AS_1 via the channel is assured to be coming from AS_2. To extend Proposition 1, we need to add two assumptions and modify another assumption.

Proposition 2 *(Two-Way Self-authentication of encrypted channels)*
Suppose:

1. *Principal P_A shares a unique secret key, K_A, with principal P_B;*
2. *Principal P_B shares the same unique secret key, K_A, with principal P_A;*
3. *P_A and P_B uses K_A to establish an encryption channel between them;*
4. *K_A^B is the key index of K_A^{-1} in P_B.*
5. *K_A^A is the key index of K_A^{-1} in P_A.*

then when P_B receives $(K_A^B, \{s\}_K)$, P_B can infer that P_A **says** *s; and when P_A receives $(K_A^A, \{s\}_K)$, P_A can infer that P_B* **says** *s.*

PROOF: Using assumptions 1, 3, 4, and Proposition 1, we can say that when P_B receives $(K_A^B, \{s\}_K)$, P_B can infer that P_A **says** s. With assumption 2, 3, 5 and Proposition 1 we can say that when P_A receives $(K_A^A, \{s\}_K)$, P_A can infer that P_B **says** s. ■

4 Conclusion

In this paper, we have proved that encrypted channels in SNG authenticate themselves. We have shown that the encrypted channels can achieve both one-way and two-way self-authentication.

In Section 3.1, Proposition 1 simply says that when we receive a message from an encrypted channel, we can infer that the principal who shared with us the channel encryption key is the one who send the message. Thus we can safely conclude that messages delivered by encrypted channels are authenticated. There is no extra steps needed to authenticate the messages received. The channel encryption key provides us with sender information. We can conclude that an encrypted channel authenticates itself.

In Section 3.2, we extended Proposition 1 to include two-way self-authentication as stated in Proposition 2. Proposition 2 was proved using a similar approach as used in proving Proposition 1. Hence we can conclude that encrypted channels in SNG are self-authenticated, either one-way or two-way depending on the setup configuration when a network joins the SNG.

Our next step is to prove the correctness of the Service Network Graph which can provide user authentication across heterogeneous networks of different administrative domain without sharing user authentication information.

References

[1] X.509 (03/00). *International Telecommunication Union ITU-T Recommendations X series.*

[2] A. Abdul-Rahman and S. Hailes. Using recommendations for managing trust in distributed systems. *Proceedings of IEEE Malaysia International Conference on Communication '97 (MICC'97), Kuala Lumpur, Malaysia*, 1997.

[3] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. *Hawaii Int. Conference on System Sciences 33 , Maui, Hawaii, January 2000*, January 2000.

[4] A. Abdul-Rahman and S. Halles. A distributed trust model. *Proceedings of New Security Paradigms Workshops 1997*, 1997.

[5] A. R. Au, M. Looi, and P. Ashley. Automated cross-organisational trust establishment on extranets. *Proceedings of the workshop on information technology for virtual enterprises, 2001*, (7):3–11, January 2001.

[6] T. Beth, M. Borcherding, and B. Klien. Valuation of trust in open networks. *Proceedings of the Conference on Computer Security 1994*, 1994.

[7] D. Denning. A new paradigm for trusted systems. *Proceedings of 1992-1993 ACM SIGSAC New Security Paradigms Workshop*, 1993.

[8] E. T. F. (IETF) and the Internet Engineering Steering Group (IESG). The kerberos network authentication service (v5). *Porpoised Standard, RFC1510.*

[9] D. Lai and Z. Zhang. Integrated key exchange protocol capable of revealing spoofing and resisting dictionary attacks. *Technical Track Proceedings, 2nd International Conference, Applied Cryptography and Network Security, Yellow Mountain*, June 2004.

[10] D. Lai and Z. Zhang. An infrastructure for service authentication and authorization revocation in a dynamic aggregation of networks. *WSEAS Transactions on Communications*, 4(8):537–547, August 2005.

[11] D. Lai and Z. Zhang. Network service sharing infrastructure: Service authentication and authorization revocation. *Proceedings of the 9thWSEAS International Conference on Communications*, July 2005.

[12] D. Lai and Z. Zhang. Secure service sharing over networks for mobile users using service network graphs. *Proceedings, Wireless Telecommunication Syposium 2006*, April 2006.

[13] D. Lai, Z. Zhang, and C. Shen. Achieving secure service sharing over ip networks. *Proceedings, ASEE Mid-Atlantic Section Spring 2006 Conference*, April 2006.

[14] D. Lai, Z. Zhang, and H. Wang. Towards an authentication protocol for service outsourcing over ip networks. *Proceedings of the 2005 International Conference on Security and Management*, (7), June 2005.

[15] B. Lampson, M. Abadi, M. Burrows, and E. Wobber. Authentication in distributed systems: Theory and practice. *ACM Transactions on Computer Systems*, 10(4):265–310, 1992.

[16] M. Montaner, B. Lopez, and J. L. Rosa. Developing trust in recommender agents. *Proceedings of the first international joint conference on Autonomous agents and multi-agent systems*, 2002.

[17] M. Reiter and S. Stubblebine. Authentication metric analysis and design. *ACM Transactions on Information and System Security*, 2(2), January 1999.

[18] S. Robles, J. Borrell, J. Bigham, L. Tokarchuk, and L. Cuthbert. Design of a trust model for a secure multi-agent marketplace. *Proceedings of the fifth international conference on Autonomous agents*, 2001.

[19] J. Thayer, J. C. Herzog, and J. D. Guttman. Strand spaces: Proving security protocols correct. *Journal of Computer Security, 1999*, 1999.