A Study on Security Techniques for Enterprise Network over Next Generation Network (NGN)

January 2011

Tadashi KAJI

梅甲 14998

A Study on Security Techniques for Enterprise Network over Next Generation Network (NGN)

Submitted to
Graduate School of Information Science and Technology
Osaka University

January 2011

Tadashi KAJI

List of Publications

Journal Papers

1. Tadashi KAJI, Seiichi SUSAKI, Katsuyuki Umezawa, Katsuhiko KONDO, Satoru TEZUKA, Ryoichi SASAKI and Akira KITO: "Seamless Object Authentication : A Security Mechanism for Mobile Codes on Distributed Object Systems," Transactions of Information Processing Society of Japan, Vol. 42, No. 3, pp.586-594(2001) (in Japanese).
2. Tadashi KAJI, Takahiro FUJISHIRO and Satoru TEZUKA: "A proposal of TLS implementation for cross certification model", IEICE Transactions on Information and Systems, Vol. E91-D, Issue 5, pp. 1311-1318 (2008).
3. Tadashi KAJI, Takahiro FUJISHIRO, Seiichi SUSAKI, Eri KAWAI, Kazuyoshi HOSHINO and Teruo HIGASHINO: "A Proposal of Secure Session Provider Service over NGN," IEEJ Transactions on Electronics, Information and Systems, Vol. 130, No. 8, pp. 1455-1463 (2010) (in Japanese).

Conference Papers

1. Tadashi KAJI, Kazuyoshi HOSHINO, Takahiro FUJISHIRO, Osamu TAKATA, Akifumi YATO, Keisuke TAKEUCHI and Satoru TEZUKA : " TLS handshake method based on SIP, " Proceedings of 1st International Workshop on Secure Information Systems SIS'06, Vol. 1, pp.467-476, ISSN 1896-7094 (October 2006).
2. Tadashi KAJI, Takahiro FUJISHIRO, and Shinichi IRUBE: "IdP proxy for combined authentication based on multiple IdPs," 4th IEEE International Symposium on Advanced Networks and Telecommunications Systems (ANTS), Mumbai, India (December 2010) (in press).

Other Related Papers

1. Tadashi KAJI, Osamu TAKATA, Kazuyoshi HOSHINO, Takahiro FUJISHIRO and Satoru TEZUKA: “A Model for Establishing Secure Communication in Secure Service Platform,” Information Processing Society of Japan, CSEC-028, pp.151-156 (2005)(in Japanese)
2. Recommendation ITU-T X.1152, “Secure end-to-end data communication techniques using trusted third party services,” ITU-T (2008) (Editor)
3. Draft Recommendation ITU-T X.sap-4, “The general framework of combined authentication on multiple identity service provider environment, ” ITU-T (2010) (Editor)

Abstract

In these days, large enterprises have enterprise networks, which consist of several local area networks (LANs) in headquarters, branch offices, and small offices or home offices (SOHOs) connected by one or more wide area networks (WANs). Although there are many types of WANs like private networks, IP-VPN or the Internet VPN at this moment, the Next Generation Network (NGN) is also considered as a cost-effective WAN, which is less expensive than private networks and IP-VPN, and it is more secure than the Internet.

NGN is an IP-based public telecommunication network, which is able to provide QoS and security to the communication between its users. To achieve these features, NGN uses Session Initiation Protocol (SIP) to control and manage communication sessions over NGN. And, one of the NGN implementation uses the network border element function linked to SIP messages to open and close a pin-hole so that communication sessions can use a priority bandwidth. Moreover, there is a service that NGN controls network access of users according to the result of authentication that an authentication server provided application service provider authenticates them. However, the security protection of NGN is limited to communications inside NGN at this time.

This thesis studies three security techniques to address security-related issues on the enterprise network over NGN: (1) the protection of communications over the enterprise network, especially, the protection of communications crossing over NGN and the Internet, (2) a TLS implementation method on cross-certification environment that can reduce a number of certificates to be managed, and (3) an object authentication method that the mobile code platform transfers and verifies a digital signature of each object code which is running on a SOHOs to protect the enterprise network from malicious coding.

The first security technique, a secure session provider service, protects communications crossing over NGN, the Internet and LANs in cooperation with the NGN's call session control function. A secure session provider service consists of a user plane function, which protects communication data, and a control plane function, which controls the user plane function in cooperation with CSCF of NGN. And a secure session provider service uses SIP as a protocol of the control plane and TLS or IPsec as a protocol of the user plane. Especially, because a secure session service provider server authenticates all of entities as a trusted third party and it skips a peer entity authentication at the communication session establishment phase, it is able to establish a secure communication session of the user plane quickly.

The second security technique aims to protect SIP messages by TLS under the interconnection with multiple public key infrastructure (PKI) environments to allow to manage a huge number of certification authorities (CAs). Once many enterprises use NGN as a part of their enterprise network, SIP intermediates in the enterprise network and NGN will be required to manage a huge number of CA certificates if enterprises and NGN service providers will operate their own PKI respectively. For the interconnection with multiple PKI environments, there are two models; "multiple trust anchors environment" and "cross certification environment". If there are many CAs, the cross certification environment is useful for such a situation. However, because TLS is designed for the multiple trust anchors model, TLS will not be able to work efficiently on the cross-certification model. This thesis proposes and evaluates a profile of TLS handshake protocol for the cross certification environment and a TLS implementation method with a delegated certification path discovery and validation server for the cross certification model efficiently.

In recent, mobile code technologies are used to help the manager of the enterprise network to maintain its application service. However, because it is difficult for a manager of an enterprise network to maintain SOHO equipments properly, malicious services running on SOHO equipments may harm the entire enterprise network. The third security technique aims to protect from the problem. Although most of existing mobile code execution platforms use digital signature technology to protect from malicious codes, they verify a digital signature of a mobile code only at time when the mobile code begins executing. Once the mobile code executed, the mobile code can access any resources in the enterprise network by the user's privilege. This thesis proposes a seamless object authentication (SOA) that a server received an access request from the mobile code verifies a digital signature of the mobile code as a countermeasure of the above security problem. If the enterprise network supports SOA mechanism, a network manager is able to control whether to allow access to the mobile code based on the digital signature. Therefore, even if a malicious mobile code is executed by careless of the user, the enterprise network is able to be protected.

The security techniques proposed in this thesis are able to achieve secure enterprise network in cooperation with NGN call control function and NGN security function.

Contents

List of Figures

List of Tables

Chapter 1
Introduction

In these days, large enterprise have enterprise networks, which consist of several local area networks (LANs) in headquarters, branch offices, and small offices or home offices (SOHOs) connected by one or more wide area networks (WANs) . Although there are many types of WANs like private networks, IP-VPN or the Internet VPN at this moment, the Next Generation Network (NGN) is also considered as a cost-effective WAN, which is less expensive than private networks and IP-VPN, and it is more secure than the Internet.

NGN is an IP-based public telecommunication network, which is able to provide quality of service (QoS) protection and security to the communication between its users. In addition, some network service providers are considering NGN as a service platform that constructs a service distribution and management framework for end-users and home service providers because they are considering that the distribution and management service of the home network services are one of important functions of NGN.

One of the important features is that NGN can provide QoS and security to the communication between its users. NGN uses call session control functions (CSCFs) and Session Initiation Protocol (SIP, [4]) to control and manage network resources for communications over NGN for QoS protection. NGN also uses Network Border Elements (NBEs) to prevent core functional components like CSCF from unauthorized access.

And also, it provided the detailed analysis on security vulnerability of NGN architecture, functional components, protocols and security management. And it suggested the key active defense strategies in network built security, boundary security, access security and information transmission security.

NGN also plans to provide security assured communications to its users as a basic network function. For example, one of NGN implementation uses NBE function linked to SIP messages to open and close a pin-hole so that communication between users whose session has be established by SIP messages can use a priority bandwidth.

Moreover, there is a service that NGN controls network access of users according to the result of authentication that an authentication server provided application service provider authenticates them.

However, at this point of time, the security provided by NGN is limited to communications inside NGN. And, all equipments (ex. CSCFs, NBEs, routers and home gateways) should be well-managed to achieve enough security.

For the enterprise network over NGN, the interconnection between NGN and the Internet will be widely used in near future. However, it will become an important problem to address how it protects the communications crossing over NGN and the Internet. In the Internet, secure communication protocols based on cryptography (ex. IPsec and TLS) are widely used to authenticate the communication peer and protect communication data. Therefore, it is considered that secure communication protocols will be used to protect communications crossing over NGN and the Internet.

When many enterprises use NGN as a part of their enterprise network, CSCFs and UAs in the enterprise network and NGN will be required to manage a huge number of CA certificates if enterprises and NGN service providers will operate their own PKI respectively. For the interconnection with multiple PKI environments, a PKI environment called as "cross certification environment" is useful to reduce the cost of certificate management. However, TLS will not be able to work efficiently on the cross-certification model because TLS is designed for another PKI environment called as "multiple trust anchors model".

Another important problem is to assure that all equipments are well-maintained. Especially, to assure functional components on a remote device like a home gateway is a difficult problem. When NGN uses as a service platform, various functional components will be downloaded and run automatically on the home gateway. By such a kind of mobile codes, the home gateway can only have basic functions (network access and component downloading) initially, and download additional components from NGN and execute them when an application is executed. To prevent from distribution of the malicious code, these mobile codes are recommended to be "signed mobile codes" that are digitally signed by a developer and/or distributor, and the home gateway executes the code if and only if the code has a digital sign by a trusted signer. However, a new security problem called as a "signed malicious code" problem arises. This problem is occurred when a malicious code is digitally signed and the home gateway mistakes the signer of the malicious code for a trusted party. Once the home gateway accepts the malicious code, it cannot be prevented by using security functions of NGN.

This thesis studies three security techniques to address security-related issues on enterprise networks over NGN: (1) the protection of communications over an enterprise network, especially, the protection of communications crossing over NGN and the Internet, (2) a TLS implementation method on cross-certification environment that can reduce a number of certificates to be managed, and (3) an object authentication method that the network platform verifies a digital signature of an object code which is running on a home gateway to protect the enterprise network from malicious coding.

Hereinafter, the existing work related with the above topics was summarized in Chapter 2. Chapter 3 describes a model of enterprise networks over NGN dealt with in this thesis and security threats to be solved. Chapter 4 proposes a secure session provider service as a first security technique. Chapter 5 proposes a TLS implementation method for the cross certification model as a second security technique. Chapter 6 proposes a seamless object authentication (SOA) method as a third security technique. Finally, Chapter 7 gives a conclusion.

Chapter 2
Related works

2.1 Virtual private network (VPN) service

In these days, various virtual private network (VPN) technologies were proposed as mentioned in [89] and [90]. And also various VPN services are offered by network service providers as wide area network (WAN) services for enterprise.

In the past, VPN services using Layer 2 technologies, such as Frame Relay and asynchronous transfer mode (ATM) were provided by network service providers. However, these Layer 2 VPN technologies have a scaling problem that the number of required virtual circuits achieving optimal routing scales non-linearly if the network grows.

To solve the scaling problem, a border gateway protocol/multiprotocol label switching (BGP/MPLS) VPN ([81], [82]) service was provided. GBP/MPLS VPN service is using BGP to carry route information in the shared network infrastructure (a MPLS core). This Layer 3 MPLS-VPN solution achieves all of the security of the Layer 2 approach, while adding enhanced scalability inherent in the use of Layer 3 routing technology. In recent, "entry" VPN services are offered ([83]). The entry VPN service is using a shared access network (like NGN).

Another alternative of Layer 2 VPN is Wide Area Ethernet (WAE, [84], [85]) that is high-speed WAN service using Ethernet connectivity. Essentially, WAE can simplify linking remote locations.

As spreading the Internet and encryption technologies, Internet VPN ([86], [87]) was provided. Internet VPN is a VPN that using encapsulating protocol (e.g. IPsec) to protect communication data over the public network (like Internet).

During current several years, entry VPN services and Internet VPN are rapidly growing in the VPN market ([88]). NGN based entry VPN services are also offered.

2.2 The next generation network (NGN)

The NGN framework is set by the International Telecommunication Union–Telecommunication Standardization Sector (ITU-T), (European Telecommunications Standards Institute), especially its Technical Committee TISPAN (TC TISPAN). Other standardization organizations such as the Internet Engineering Task Force (IETF), Third Generation Partnership Project (3GPP), American National Standards Institute (ANSI), ECMA International (ECMA) and Open Mobile Alliance (OMA) are actively involved in defining NGN standards.

ITU-T has discussed an NGN in standards since at least 2003 and published a series of NGN Release 1 Recommendations. ITU-T defined a scope ([67] and [68]), a general overview of NGN ([62]), terms and definitions ([66]), a capability set ([63]) and functional requirements and architecture ([64]). ITU-T also developed a Network attachment control functions (NACF, [65]). NGN standardization activities in ITU-T were surveyed in [59], [60] and [61].

NGN is defined in [62] as follows: "*A packet-based network able to provide telecommunication services and able to make use of multiple broadband, QoS-enabled transport technologies and in which service-related functions are independent from underlying transport-related technologies. It enables unfettered access for users to networks and to competing service providers and/or services of their choice. It supports generalized mobility which will allow consistent and ubiquitous provision of services to users.*"

On the above definition, the separation between services and their transport and the addition of quality of service (QoS) to IP-based transport are recognized as significant features of NGN.

ETSI TC TISPAN published NGN Release 1, which provides the first set of implement for NGN specifications. ETSI TC TISPAN developed a functional architecture ([70]),the Network Attachment Subsystem (NASS, [71]), which provides registration at the access level and initializes terminal accessing to NGN services, and the Resource and Admission Control Subsystem (RACS, [72]), which provides applications with a mechanism for requesting and reserving resources from the access network. And also, ETSI TC TISPAN has adopted the IMS ([73]). NGN standardization activities in ETSI were surveyed in [69].

2.3 Security of NGN

There are two types of security studies in related to NGN: security for NGN and NGN for security.

The study of security for NGN is a study to protect NGN from certain security threat(s).

The study of NGN for security is a study to provide security service by NGN functions (include NGN security functions).

2.3.1 Studies of security for NGN

For security of NGN, ITU-T issued Recommendation Y.2701 ([8]). This Recommendation describes a method to achieve security by using the trust model, that is, an NGN composed of trusted domain, un-trusted domain, and trusted but vulnerable domain in-between. One of the key issues to achieve security with this model is the method to transmit signaling, media, OAMP (Operation, Administration, Maintenance and Provision) traffic from the un-trusted domain to the trusted domain.

ETSI TC TISPAN also issued the security requirements pertaining to TISPAN NGN in [7]. This described requirements for the various NGN subsystems and covered security requirements for both the NGN core network, and the NGN access network(s).

ECMA issued a series of technical reports ([75] [76]) which explore IP-based enterprise communication from/to Corporate telecommunication Networks including. Ref. [75] describes 5 models of the next generation corporate network (NGCN)-NGN interconnection and listed 48 requirements to NGN, which includes 24 security-related requirements.

Nisase presented basic ideas for achieving security in certain NGN implementation in [17]. In this NGN implementation, blocking packets from/to private address spaces using an NBE-equivalent function limits reachability from points outside NGN unnecessary for its services to points within the NGN to reduces unauthorized access across the entire network. This NGN implementation also uses an NBE-equivalent function to prevent spoofing-related attacks that misrepresent the source IP (Internet protocol) address or originating telephone number.

Zhang et al. described a detailed analysis on security vulnerability of NGN in [9]. They also discussed possible attacks by hijack, concluded NGN attacks and proposed an active defense strategy in network-built security, boundary security, access security and information transmission security.

Khan et al. proposed an Intrusion Detection and Prevention (IDP) system for IP Multimedia Subsystem based multiparty conference system in [10]. This IDP is to mitigate the effect of signaling message (ex. SIP INVITE and SIP REGISTER) flooding attacks and internal threats. They also proposed an election based distributed referring authority mechanism to handle the conflict arise due to the single participant based referring.

For security of IP Multimedia Subsystem (IMS), Awais et al. analyzed the security vulnerabilities and requirements of IP Multimedia Subsystem (IMS), particularly the impact of Denial-of-Service (DoS) and Distributed DoS (DDoS) attacks on the IMS in [11]. They also proposed an intelligent Bio-inspired, self-defending security framework for the IMS and Next Generation all-IP Networks.

2.3.2 Studies of NGN for security

Some network service providers are considering NGN as a service platform that constructs a service distribution and management framework for end-users and home service providers because they are considering that the distribution and management service of the home network services are one of important functions of NGN ([1]).

Nisase also presented a particularly robust defense function for SIP-controlled session-based communications that have quality and reliability requirements in [17]. In this NGN implementation, an NBE-equivalent function linked to SIP messages is used to open and close a pinhole so that communications between terminals whose session has been established by SIP messages can use a priority bandwidth.

In addition, certain NGN implementation provides a service, which controls session establishment based on the authentication result by an authentication server located outside NGN (i.e., in the application service provider network) ([6]).

Fukazawa et al. proposed that multiple profiles should be used by NGN providers in [2]. They also proposed a strong authentication mechanism and an easy peer-to-peer broadband virtual private network service as service and application examples.

Kawashima et al. proposed the SIP DIAL-UP method in [5]. The SIP DIAL-UP method is controlling Internet Key Exchange Protocol (IKE) and IPsec connection with SIP signaling for establishing remote access by using the NGN CSCF and RACF capabilities. The overviews of the SIP DIAL-UP method are described in Appendix A.

2.4 Security of mobile code

In recent information systems, security function becomes one of the most important functions. Distributed object systems and mobile code systems have many security functions.

2.4.1 Security functions of existing distributed object system

Some of existing distributed object systems ([27][29]) have following security functions:

- Peer entity authentication function: it is a function to confirm the assurance of the claimed identity of principals (like users or objects).
- Access control function: it is a function to limit the access to the system resource only to the authorized principals.
- Security audit function: it is a function to make audit trails to ensure compliance with established policy and operational procedures, to detect breaches in security, and to recommend any indicated changes in control, policy and procedures.

- Communication protection function: it is a function to protect exchanged data with peer entity to prevent from eavesdropping, injection and modification of data and unauthorized access.
- Non-repudiation function: it is a function to make a recipient of data with proof of the origin of data and/or with proof of delivery of data.
- Security management function: it is a function for administrators to maintain security policies to control the security-related behavior of the system.

2.4.2 Security functions of existing mobile code system

Many mobile code systems have security functions in nature ([23][24][25] [28]). Some mobile code systems implements security functions based on a security model called as "sand-box" model characterized as follows:

- It is not allowed for a mobile code to access to local resources.
- It is not allowed for a mobile code to access to other servers except for the server in which the mobile code is located.

Although the "sand-box" model is suitable for achieving stronger security in the mobile code system, the above strong limitations also ruin a flexibility of the system.

For the balance of security and flexibility of the mobile code system, an extension called as "signed" mobile code is introduced. The extension is characterized as follows:

It is allowed for a mobile code to access local resources and other servers if the mobile code is a digitally signed and the signer is trusted by the user.

The "signed" mobile code can be used to confirm who is a developer/distributor of the mobile code and to prevent from unauthorized modification of the mobile code.

Chapter 3
Models of enterprise network over NGN

This chapter describes a model of enterprise networks over the next generation network (NGN) at first. And then, threats to be solved are described.

3.1 Basic model of enterprise network over NGN

In the recent enterprise, an enterprise network consists of several LANs in a headquarters, several branch offices and several SOHOs and WANs which are connected these LANs each other.

Figure 3.1 shows a model of enterprise network over NGN dealt with in this thesis. This model is extension of model 2 and 4 described in [76] and supports not only telecommunication service but also data communication service.

It notes that model 2 and 4 can also apply to not closely related partners, which share no common security policy and no security tunnel. Therefore, the model shown on Figure 3.1 can also support SOHOs which are managed by third party. The model shown in Figure 3.1 consists three types of LANs (in headquarters, branch office and SOHO), NGN connected with LANs in headquarters and branch offices and the Internet connected with NGN and LAN in SOHO.

NGN-equivalent call session control function (CSCF), which is to establish QoS-enabled communication sessions in cooperate with CSCF in NGN, and transport function (TF) are located in the headquarters' LAN. (CSCF in the headquarters LAN establish QoS-enabled communication sessions according to security policies defined and managed in headquarters. If SOHOs are managed by third party, identity federation can be used for mapping between securities policy of the enterprise and SOHOs.)

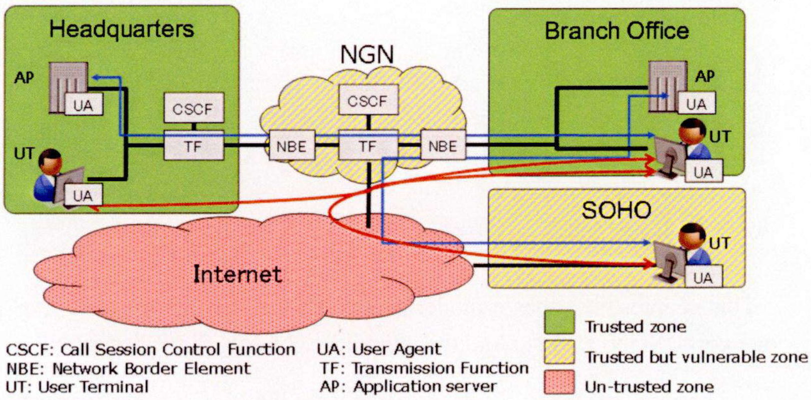


Figure 3.1 Model of enterprise networks over NGN

In this model, each terminal (user terminals (UT) and application server (AP)) acts as a user agent (UA).

From the viewpoint of the enterprise network manager, LANs of headquarters and branch offices are trusted, NGN and SOHO are trusted but vulnerable, and the Internet is un-trusted.

On the above model, there are two types of communications; client-server communication (blue line in Figure 3.1) and P2P communication (red line in Figure 3.1). The client-server communication is a communication between application servers, in the LAN in headquarters or branch offices and users in the headquarters, branch offices or SOHOs. The P2P communication is a communication between users.

3.2 Threats of enterprise network over NGN

As mentioned above, NGN and SOHO are trusted but vulnerable, and the Internet is un-trusted. It means that equipment in NGN and SOHO and data transmitted in NGN, SOHO and the Internet are exposed to the threat from the Internet directly.

For the security of communications on the above basic model, threats shown in Table 3.1 are required to be considered.

Table 3.1 Threats to be considered

	Threat	Target	Place
1	Eavesdropping	communication data	in NGN
2			in the Internet
3			in SOHO
4		signaling data	in NGN
5			in the Internet
6			in SOHO
7	Destroy/ modification	communication data	in NGN
8			in the Internet
9			in SOHO
10		signaling data	in NGN
11			in the Internet
12			in SOHO
13	Impersonation	headquarters user	from the Internet
14		branch office user	from the Internet
15		SOHO user	from the Internet
16	Phishing	headquarters user	from the Internet
17		branch office user	from the Internet
18		SOHO user	from the Internet
19	Unauthorized access	UT or AP in headquarters	from the Internet
20		UT or AP in branch office	from the Internet
21		UT in SOHO	from the Internet
22	Information leakage	from UT or AP in headquarters	to the Internet
23		from UT or AP in branch office	to the Internet
24		from UT in SOHO	to the Internet

It notes that these threats in Table 3.1 may come not only from the Internet but also from SOHO if some SOHO environments are not managed properly.

Because NGN has security function in nature, some security threats are addressed by NGN. Namely, security threats "in NGN" can be addressed by NGN. However, security threats "in the Internet" and "from the Internet" cannot be addressed only by NGN security function.

And the security threat for unauthorized access from the Internet to headquarters or branch offices can be addressed by NGN NBE function. And the security threat for unauthorized access from the Internet to SOHOs may be addressed by traditional firewall if SOHOs are managed properly.

And also, NGN security function can address security threats for information leakage from headquarters to the Internet "via NGN directly" and information leakage from branch office to the Internet "via NGN directly" if NGN inspects communication between the headquarters (or branch offices) and the Internet. However, NGN security function can address security threats for information leakage from headquarters to the Internet "via SOHO user" and information leakage from branch office to the Internet "via SOHO user" because SOHO is connected to the Internet directly. The detail and proposed solution of this problem is described in Chapter 6.

As shown in Table 3.2, the following threats are required to be addressed for secure enterprise network:

(1) Eavesdropping of communication data in the Internet and SOHO (i.e., non-NGN network section),

(2) Modification/Destroy of communication data in non-NGN network section,

(3) Impersonation from the Internet,

(4) Phishing from the Internet,

(5) Unauthorized access from the Internet to UT in SOHO and

(6) Information leakage to the Internet

Table 3.2 Threats to be addressed

	Threat	Target	Place	Protected by NGN
1	Eavesdropping	communication data	in NGN	Yes
2			in the Internet	No
3			in SOHO	No
4		signaling data	in NGN	Yes
5			in the Internet	Yes
6			in SOHO	No
7	Destroy/ modification	communication data	in NGN	Yes
8			in the Internet	No
9			in SOHO	No
10		signaling data	in NGN	Yes
11			in the Internet	Yes
12			in SOHO	No
13	Impersonation	headquarters user	from the Internet	No
14		branch office user	from the Internet	No
15		SOHO user	from the Internet	No
16	Phishing	headquarters user	from the Internet	No
17		branch office user	from the Internet	No
18		SOHO user	from the Internet	No
19	Unauthorized access	UT or AP in headquarters	from the Internet	Yes
20		UT or AP in branch office	from the Internet	Yes
21		UT in SOHO	from the Internet	No
22	Information leakage	from UT or AP in headquarters	to the Internet	No
23		from UT or AP in branch office	to the Internet	No
24		from UT in SOHO	to the Internet	No

Chapter 4
Secure Session Provider Service over NGN

4.1 Introduction

This chapter proposes a secure session provider service that is a system against the problem about how it protects communications crossing over NGN and the Internet.

The secure communication provider service sets up an encrypted communication sessions between a communication device on NGN and a communication device on the Internet between up in cooperation with NGN's call session control function (CSCF) to protect communications crossing over NGN and the Internet. Especially, the secure communication provider service can be used not only for communications between a server and a client, but also for peer-to-peer (P2P) communications.

4.2 Requirements of the secure session provider service

4.2.1 Application model

Figure 4.1 shows two types of application models of the secure session provider services. One is a client-server model for protection of communications between an application server and a user terminal, and another one is a P2P model for protection of P2P communications between user terminals.

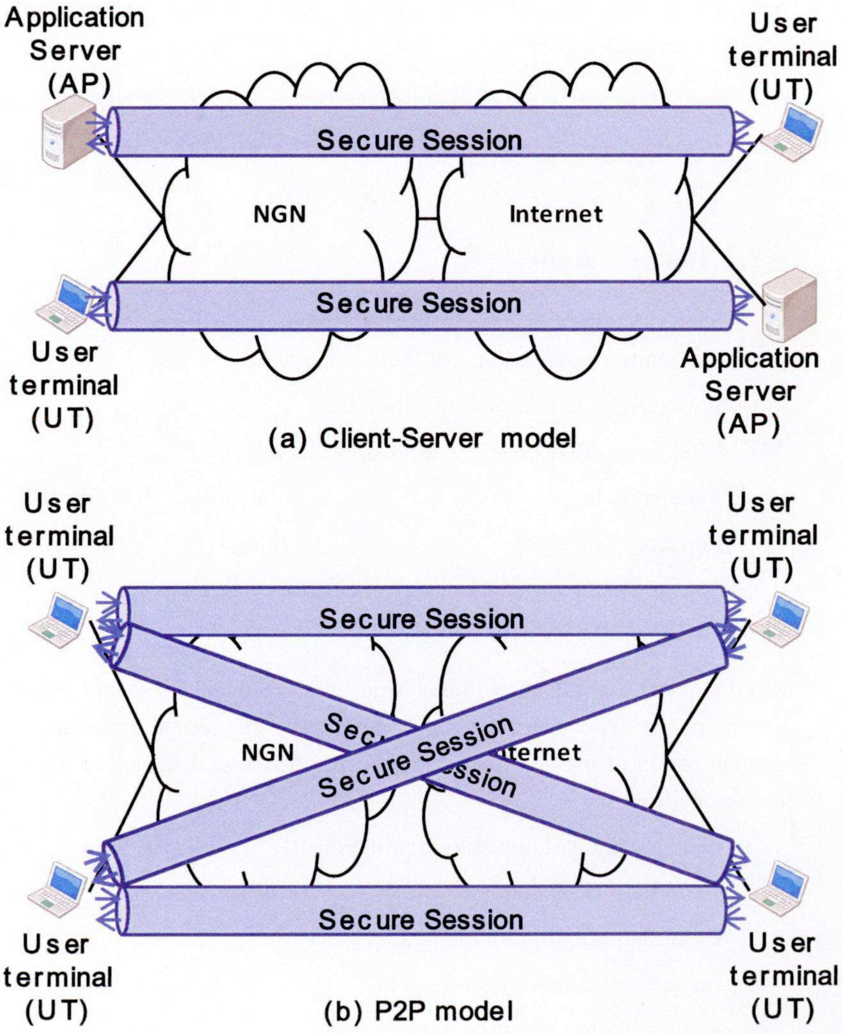


Figure 4.1 Application model

The client-server model is to communicate between an application server (AP) on the Internet and a user terminal (UT) on NGN or between an AP on NGN and UT on the Internet. This mode is for each NGN user to retrieve some information from AP via UT or to use a contents delivery service (AP) on NGN from UT on the Internet.

The P2P model is to communicate between UTs on NGN and/or on the Internet. This model is for file sharing services or messaging services between users.

4.2.2 Threats model

On the above application models, the secure communication provider service provides countermeasures against the following four threats:

- Eavesdropping of communication data
- Modification/Destroy of communication data
- Impersonation
- Phishing

4.2.3 Functional requirements

From the above application models and threats model, the secure communication provider service provides an encryption of communication data and authentication of the communication peer and also is designed to meet following requirements:

- Independency of application protocol
- Fast establishment of encrypted communication session
- Confidentiality of communication data
- Integrity of communication data
- Authentication of communication peer

(1) Independency of application protocol

As mentioned above, there are two types of communications crossing over NGN and the Internet: communication between a server and a client and peer-to-peer communication. In addition, various application protocols are used on these two types of communications.

Therefore, the secure communication provider service is required to protect communications independent with communication types and/or application protocols.

(2) Fast establishment of encrypted communication session

There are some application services required to establish many communication sessions with multiple communication peers like P2P communication or service mush-up. For such a kind of application services, it is required to establish an encrypted communication session quickly to minimize impact against user experiences.

(3) Confidentiality of communication data

It is required to protect confidentiality of communication data to prevent from eavesdropping of communication data.

(4) Integrity of communication data

It is required to protect integrity of communication data to prevent from modification or destroy of communication data.

(5) Authentication of communication peer

It is required to authenticate the communication peer to prevent from impersonation and phishing.

4.3 Secure session provider service

4.3.1 Overview of the authentication mediation model

This thesis proposes an authentication mediation model as a base model of the secure communication provider by using the call session control function of NGN.

Table 4.1 Planes of secure communication

Plane	Functions
User plane	Protection of communication data confidentiality
	Protection of communication data integrity
Control plane	Peer entity authentication
	Authorization (access decision)
	Negotiation of security association
	Error/Alert notification
	Acknowledgement
	Revocation of security association

As shown in Table 4.1, the secure session consists of two planes: the user plane and control plane. The user plane is to protect application data and the control plane is to control the user plane.

As mentioned in [92], traditional VPN models are classified into two models: full-mesh VPN model and hub-and-spoke VPN model.

In full-mesh VPN, one communication device is establishing user planes and control planes with other devices directly and any two communication devices perform all process including peer authentication and security association generation.

In hub-and-spoke VPN ([91]), one communication device is establishing user plane and control plane with one hub device and all of data exchanged between any two communication devices are decrypted and re-encrypted by the hub device.

The authentication mediation model introduces a trusted third party (TTP) which controls control planes and performs peer authentication and security association generation on behalf of communication devices to mediate establishment of encrypted communication sessions. In the authentication mediation model, one communication device is also establishing user planes with other devices directly. However, one communication device is establishing only one control plane with TTP. Because application data are exchanged between any two communication devices directly, TTP in the authentication mediation model needs less computing power than the hub device in the hub-and-spoke VPN system. And also, because TTP performs peer authentication and security association generation on behalf of communication devices, communication devices in the authentication mediation model need less computing power than communication devices in the full-mesh VPN system.

Figure 4.2 shows the overview of the authentication mediation model. To establish a secure communication session between APs and/or UTs, the authentication mediation model consists of three phases: the authentication phase, the key distribution phase and the secure data exchange phase.

At first, the authentication phase is performed. In the authentication phase, TTP authenticates UTs and APs respectively. If authentications are success, TTP establishes and keeps control plane sessions with UTs and APs. And also, UTs and APs register their security policies (SPs) to TTP via the established control plane sessions.

Next, the key distribution phase is performed. In the key distribution phase, TTP generates a security association (SA) for the secure communication session between AP and UT. And then, TTP distributes SA to AP and UT through control plane sessions. This phase is triggered when UT (AP) tries to communicate with AP (UT).

And then, the secure data exchange phase is performed. This phase follows the key distribution phase immediately. At the secure data exchange phase, UT and AP confirm if they share same SA or not. And UT and AP start the application data transmission through the established secure communication session.

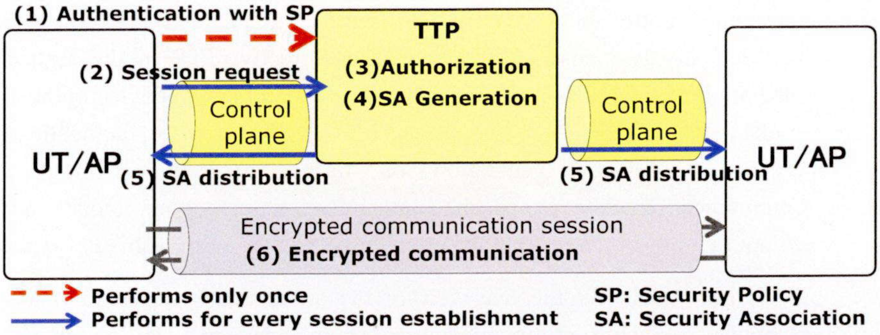


Figure 4.2 Basic concept of the proposed method

In the authentication mediation model, peer authentication is performed before all of secure communication session establishments. As the result, the authentication mediation model can establish encrypted communication sessions effectively and quickly.

In the authentication mediation model, there are other two phases: key de-registration phase and logout phase.

The key de-registration phase is a phase to terminate a secure communication session between AP and UT. After finishing the secure data exchange phase, the UT (AP) requests to TTP. TTP sends a message to delete an SA to AP and UT through control plane sessions. When AP and UT receive the message from TTP, AP and UT delete specified SA.

The logout phase is a phase for UT (AP) to logout from TTP. In the logout phase, UT (AP) sends a message to logout from TTP. When TTP receives the message, TTP deletes SP and terminates the control plane session with UT (AP).

4.3.2 Overview of the secure communication provider service

The secure communication provider service is an implementation of TTP in the authentication mediation model shown in Figure 4.2. Figure 4.3 shows a system model of a secure session communication provider service uses Session Initiation Protocol (SIP) as a control protocol of the control plane and plays as a TTP in cooperation with CSCF of NGN for encrypted communication (IPsec) sessions crossing over NGN and the Internet.

sSCP in Figure 4.3 consists of four functions and two databases: SIP user agent function (SIP UA), security policy management function (SPM), back to back user agent function (B2B UA), Third party call control user agent function (3PCC UA), policy information database (Policy Info. DB) and session information database (Session Info. DB).

SIP UA is a function to send or receive SIP messages and thereby manage a SIP session with UT or AP. And also, when establishing a SIP session, SIP UA authenticates UT (or AP).It notes that the SIP session between SIP UA and CSCF is protected by IPsec or TLS.

SPM is a function to manage SPs stored in the Policy Info. DB and to create a security association (SA) from two security policies by selecting common element between two SPs. SA created by SPM is distributed to UT and AP.

B2B UA is a function to handle the SIP message exchange to store SPs in the policy information database.

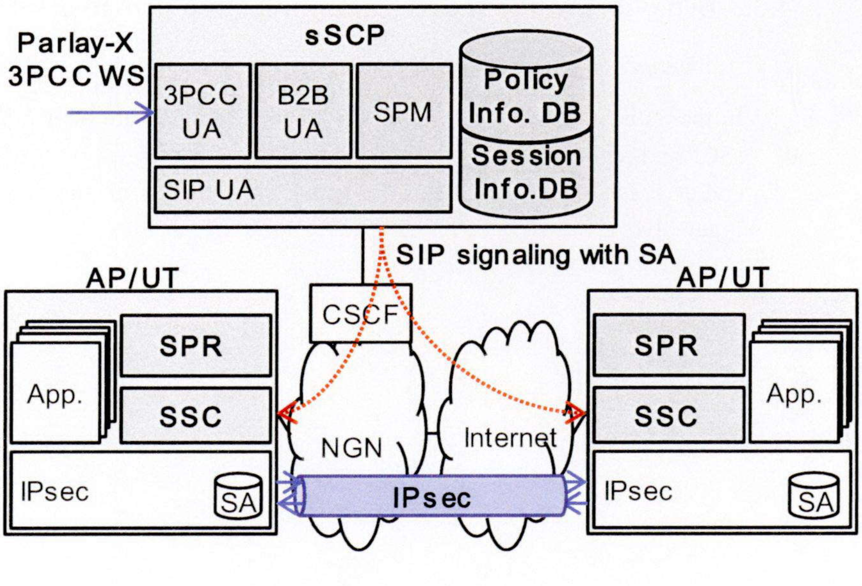


Figure 4.3 System architecture

3PCC UA is a function to receive requests to establish or terminate an IPsec session as Parlay-X Third Party Call Web Service (Parlay-X 3PCC WS) ([16]).

Policy information database is a database to store SPs with SIP-URI.

Session information database is a database to store IPsec session information with two SIP-URI and Call-ID (defined in [4]).

In addition, UT and AP in Figure 4.2 consist of three functions and one database to provide communication data protection to applications (Apps): security policy register function (SPR), secure session control function (SSC), IPsec communication function (IPsec) and security association database (SADB).

SPR is a function to maintain its own SP.

SSC is a function to communicate with sSCP by SIP protocol and to store SA in SADB (or delete it from SADB). It notes that the SIP session between SIP UA and CSCF is protected by IPsec or TLS.

IPsec is a function to protect application data by using SA stored in SADB.

SADB is a database to store SA received from sSCP.

4.3.3 Behaviors of the secure communication provider service

(1) Authentication phase

In the authentication phase, all of APs and all of UTs are authenticated by sSCP and register their SPs on sSCP before any communication between APs and/or UTs. In detail, sSCP, APs and UTs exchange messages shown in Figure 4.4.

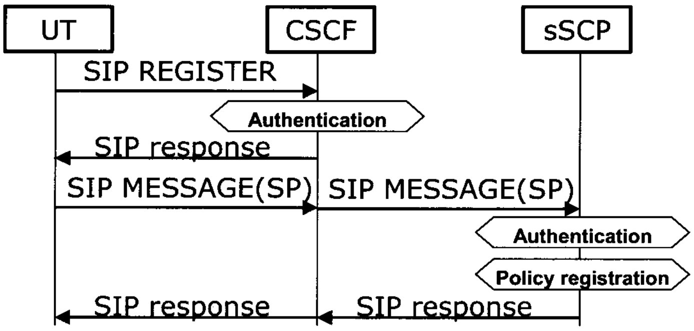


Figure 4.4 Authentication & policy registration

At first, UT (or AP) establishes a secure session (IPsec session or TLS session) with CSCF and sends SIP REGISTER message. When CSCF receives SIP REGISTER message, CSCF checks if a sender information (SIP-URI) of SIP REGISTER message is the entity, who is authenticated when the secure session was established, or not. And CSCF stores a pair of SIP-URI and IP address and returns SIP response message to UP (AP).

And then, UT (AP) sends sSCP a SIP MESSAGE message (defined in [18]), which carries with the security policy (SP).

It notes that if SOHOs are managed by third party, the SIP MESSAGE message is recommended to be digitally signed and carry with a digital certificate of the sender to make sSCP authenticate the sender directly.

When sSCP receives the message, sSCP checks a sender's information (SIP-URI) to identify the sender and verifies a digital signature of the message and the digital certificate to authenticate a sender of the message if the message is digitally signed. If authentication is success, sSCP stores SP into the policy information database.

Hereinafter, sSCP considers that the senders of messages are the same if SIP-URIs are the same because NGN prohibits from impersonation of SIP-URI.

(2) Key distribution phase

After registration of SP of a UT, the UT can be allowed to establish IPsec sessions with APs (or other UTs). The key distribution phase is to establish an IPsec session with an AP (or UT). In the key distribution phase, the UT requests sSCP to establish an IPsec session with AP.

In detail, sSCP, APs and UTs exchange messages shown in Figure 4.5. There are two types of method to establish an IPsec session. One is a method initiated by SIP INVITE request message (defined in [4]) and another is a method initiated by Parlay-X 3PCC WS.

In the method initiated by SIP INVITE request message, a UT sends sSCP a SIP INVITE request message, which indicates an AP's SIP-URI in its body, at first.

When sSCP receives the SIP INVITE request message from the UT, sSCP makes an access decision. (i.e., the UT has a right to communicate with the AP.) And if the UT has the right, sSCP retrieves SPs of the UT and the AP from the policy information database and creates one security association (SA) defined in Table 4.2 by selecting common element between two SPs.

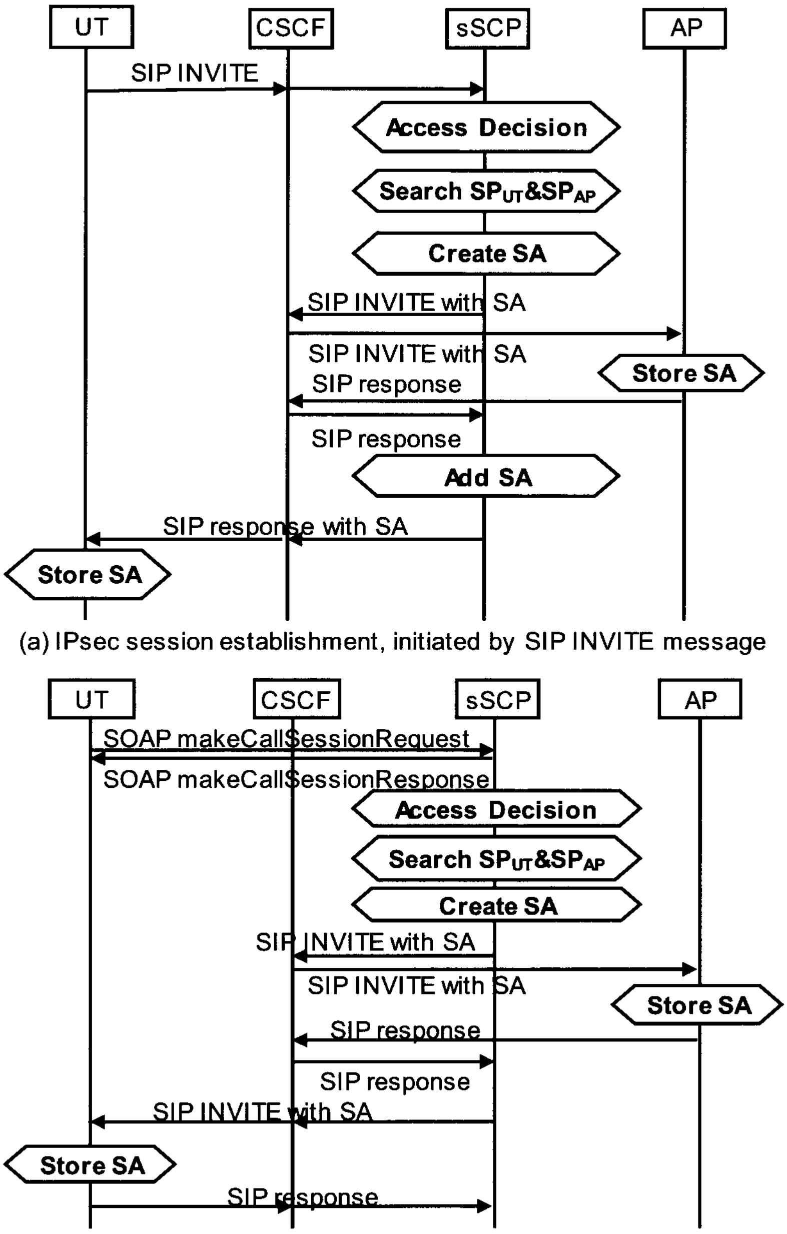
And sSCP sends the AP another SIP INVITE request message with the created SA. The AP stores SA into AP's SA DB if it is acceptable and returns a SIP INVITE response message.

When sSCP receives the SIP INVITE response message from the AP, sSCP sends the SIP INVITE response message with the SA to the UT if the received response from the AP is a positive response.

The UT stores SA into UT's SA.

As the result, the UT and the AP shares the same SA. That is, an IPsec session between the UT and the AP is established.

On the other hand, in the method initiated by Parlay-X 3PCC WS, a UT sends sSCP the SOAP message, which is called as a makeCallSessionRequest message (defined in [16]), instead of the SIP INVITE request message.



(a) IPsec session establishment, initiated by SIP INVITE message

(b) IPsec session establishment, initiated by Paray-X WS

Figure 4.5 IPsec session establishment

Table 4.2 Structure of Security Association

Component	Explanation
SPI	An identifier of this security association.
Sender information	Information in regard to the sender on this IPsec session, which contains: - IP address - port number.
Receiver information	Information in regard to the receiver on this IPsec session, which contains: - IP address - port number.
Encryption algorithm	Selected encryption algorithm to be used on this IPsec session
Encryption key	A key value to be used for encryption/decryption of communication data on this IPsec session.
MAC algorithm	Selected MAC algorithm to be used on this IPsec session.
Hash key	A key value to be used for generation/verification of message authentication code on this IPsec session.
Validity	Validity period of this security association.

When sSCP receives the makeCallSessionRequest message from the UT, sSCP makes an access decision (i.e., it checks whether the UT has a right to communicate with the AP or not) . And if the UT has the right, sSCP creates a SA from UT's SP and AP's SP.

And then sSCP, UT and AP exchange SIP messages according to SIP 3PCC message sequence ([19]) to establish an IPsec session. In detail, sSCP creates a SIP INVITE request message with the created SA and sends it to the AP. The AP stores SA into AP's SA DB and returns a SIP INVITE response message.

When sSCP receives the SIP INVITE response message from the AP, sSCP sends the SIP INVITE response message with the SA to the UT. The UT stores SA into UT's SA.

(3) Key de-registration phase

In the key de-registration phase, a UT (or an AP) sends sSCP a message to request to terminate an IPsec session with an AP (a UT). As the key distribution phase, there are two types of methods to terminate the session shown in Figure 4.6: a method initiated by SIP BYE request message and a method initiated by Parlay-X 3PCC WS.

In the method initiated by SIP BYE request message, a UT sends sSCP a SIP BYE request message, which indicates the IPsec session with the AP as a Call-ID, at first.

When sSCP receives the SIP BYE request message from the UT, sSCP makes an access decision. (i.e., the UT has a right to terminate the communication with the AP.) And if the UT has the right, sSCP retrieves corresponding session information from Session Info. DB and deletes it. DB and sends the AP another SIP BYE request message, which indicates the IPsec session with the UT as a Call-ID.

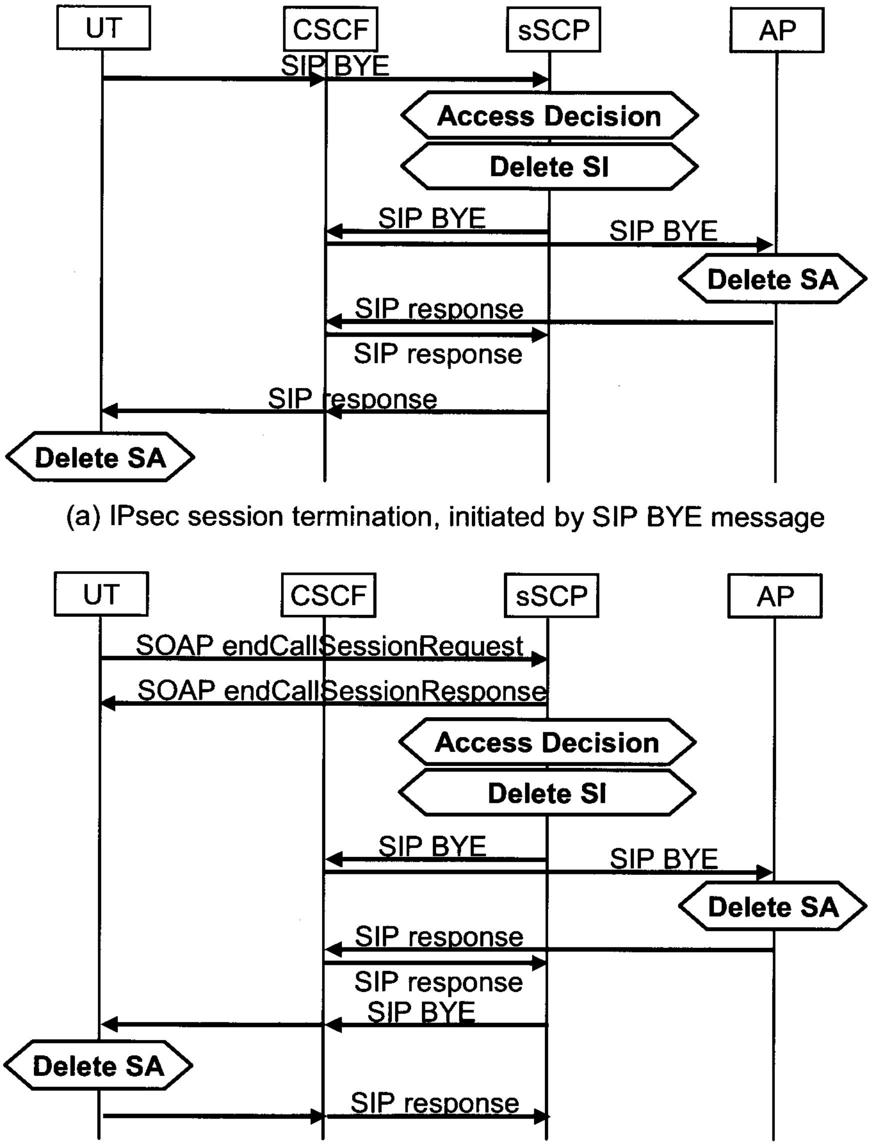
The AP deletes SA with the UT from SA DB and returns a SIP BYE response message.

When sSCP receives the SIP BYE response message from the AP, sSCP sends the SIP BYE response message to the UT. The UT deletes the SA with the AP from SA DB.

As the result, the shared SAs are deleted from the UT and the AP. That is, the IPsec session between the UT and the AP is terminated.

On the other hand, in the method using Parlay-X 3PCC WS, a UT sends sSCP the SOAP message, which is called as a endCallSessionRequest message (defined in [16]), instead of the SIP BYE request message.

When sSCP receives the endCallSessionRequest message from the UT, sSCP makes an access decision. (i.e., the UT has a right to communicate with the AP.) And if the UT has the right, sSCP deletes session information from



(a) IPsec session termination, initiated by SIP BYE message

(b) IPsec session termination, initiated by Paray-X WS

Figure 4.6 IPsec session termination

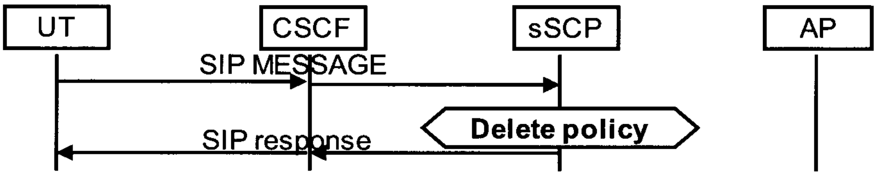


Figure 4.7 Logout

the session information database and sends the SIP BYE request message to the AP and the UT according to SIP 3PCC message sequence to terminate the IPsec session.

(4) Logout phase

In the logout phase, AP (or UT) sends a SIP MESSAGE message with no body to request sSCP to delete registered SP at the authentication phase. When sSCP receives the SIP MESSAGE message with no body, sSCP deletes SP of the sender of the message (Figure 4.7) .

4.4 Evaluation

4.4.1 Conformity of functional requirements

This section describes how the secure communication provider service satisfies functional requirements shown in Section 4.2.3.

(1) Independency of application protocol

The secure communication provider service is to set up an IPsec session and NGN is an IP based network. Therefore, the secure communication provider service can protect any kind of uni-cast application protocols like HTTP, FTP, RTP and so on.

Although the secure communication service provider in this thesis does not support to protect IP multicast at this time, it will be able to support IP multicast if a group management function is designed to manage SIP-URIs of UTs and APs which performs IP multicast communication because the secure communication service provider does not use a key exchange protocol for uni-cast communication like IKE.

(2) Fast establishment of encrypted communication session

The secure communication provider service performs costly processes (like password input, public key encryption processing and so on) in advance.

In addition, sSCP generates SAs on behalf of UTs and APs and distributes it them.

As the result, when establishing an IPsec session between a UT and an AP, the UT and the AP are not required to perform costly processes (authenticate the peer and negotiates an SA). Therefore, it is able to establish an IPsec session quickly.

(3) Confidentiality of application data

In the proposed system, confidentiality of application data is achieved by IPsec.

However, there is a threat that an attacker can decrypt an IPsec packet by acquiring an SA illegally.

There are two types of acquisition timing; when the SA is generated on sSCP and when the SA is distributed from sSCP.

To prevent from the former illegal acquisition, sSCP is required not to hold SA (especially secret information like encryption key, message authentication key and so on). Therefore, sSCP abandon an SA as soon as distribution of it even if sSCP generate the SA.

To prevent from the latter illegal acquisition, sSCP is required to protect communication data with UTs and APs. Because communication within NGN is protected by NGN, the problem is how to protect SIP messages between CSCF and UT (or AP) on the Internet. IMS, which is a specification of CSCF, is required to protect communication data between CSCF and the communication terminal by IPsec. Therefore, the confidentiality of SIP messages between CSCF and UT (or AP) is achieved by IPsec.

As the result, it is difficult to acquire SA illegally on the proposed system and the proposed system achieves the confidentiality of application data.

(4) Integrity of application data

As the confidentiality of application data, the integrity of application data is achieved by IPsec. And the SA is protected as mentioned above.

Therefore, the proposed system achieves the integrity of application data.

(5) Authentication of communication peer

In the proposed system, sSCP authenticates all of UTs and APs by digital certificate when they register SPs to sSCP. Once sSCP authenticates them, sSCP keeps pairs of SIP-URI of the SIP message and SP into policy information database.

Hereinafter, sSCP considers that the sender of SIP message is already authenticated if the sender of SIP message is already stored into policy information database.

Because NBE of NGN has a function to protect attacks from unauthenticated peers, NBE prevent from transmitting SIP messages from unauthorized user. Therefore, NGN assures the sender of SIP message is not impersonated.

In addition, sSCP sends SIP messages to SIP-URIs stored into policy information database only. Because all of SIP-URIs stored into policy information database is authenticated, sSCP can assure the sender of SIP message is not impersonated.

As the result, the proposed system prevents from unauthorized communication by the impersonation of UTs (or APs) and from phishing.

However, there is a possibility that sSCP cannot receive a message for the Logout phase even if a UT sends the message because of network troubles. In this case, because a pair of SIP-URI and SP of the UT is not deleted from policy information database, unauthorized user can communicate with APs as the authorized user. To prevent from such a problem, sSCP is required to retrieve user status information form a presence function (user management function) of NGN and update the policy information database when certain user's status is changed.

4.4.2 Comparison with existing methods

From the viewpoint of elapsed time for establishing an encrypted communication, user authentication and SA sharing are important factors because these require user interaction (like entering a password), (public key) encryption processing and message exchange. And also, these are important factors from the viewpoint of processing power because (public key) encryption processing is required much processing power.

In addition, from the viewpoint of information management, reducing authorization (access decision) points is an important factor because it requires distributing and managing access control information securely.

From above three viewpoints, comparisons of the proposed methods with existing methods (IKE and SIP DIAL-UP) are described. Especially, comparisons of processing power and of information management are described on following two communication models; the Client-Server model, which supposes that *n* UTs communicate with *m* APs respectively, and the P2P model, which supposes that *n* UTs communicate with each other.

(1) Authentication and negotiation in IKE method

In IKE method, one IKE session between two entities is required to communicate with them by IPsec. And one mutual authentication is performed when one IKE session is established.

And also, negotiation of SA is performed when one IPsec session is established.

From the viewpoint of information management, IKE method requires distributing and managing an ID of communication peer on the entity for authorization.

(2) Authentication and negotiation in SIP DIAL-UP method

In SIP DIAL-UP method, one IKE session between two entities is also required to communicate with them by IPsec. And one mutual authentication is performed when one IKE session is established.

In addition, SIP DIAL-UP method requires establishing a SIP session with CSCF before establishing any IKE session. It means that one more mutual authentication (pre-authentication) is required between the entity and CSCF.

And also, negotiation of SA is performed when one IPsec session is established.

From the viewpoint of information management, SIP DIAL-UP method requires distributing and managing an ID of communication peer on the entity for authorization.

(3) Authentication and negotiation in the proposed method

In the proposed method, no mutual authentication is performed between two entities because the entity considers that CSCF and sSCP transmits SIP messages from authenticated entity to authenticated entity.

To assure above assumption, the proposed method requires establishing a secure SIP session with CSCF. Therefore, pre-authentication is required between the entity and CSCF. In the case that the SIP MESSAGE message sent to sSCP is digitally signed, one more pre-authentication is performed.

And also, the proposed method requires no negotiation because SA is generated on sSCP and distributed to two entities via SIP sessions.

From the viewpoint of information management, the proposed method does not require distributing and managing an ID of communication peer on the communication device because sSCP performs authorization on behalf of communication devices and distributes SA to communication devices if and only if authorization is success.

(4) Comparison of elapsed time

From the viewpoint of elapsed time for establishing a secure communication session, existing two methods take time for processing authentication (entering a PIN, verification of digital signature, message exchange with AS and so on) and time for processing negotiation (message exchange with communication peer, key calculation and so on) when establish a secure communication session, but the proposed method takes only time for key distribution from sSCP because it performs *no* authentication and *no* negotiation when establish a secure communication session,

Therefore, the proposed method takes less time than existing two methods.

(5) Comparison of processing power on the Client-Server model

In IKE method, one UT establishes an IKE session and an IPsec session with certain AP. It means that one UT performs *1* authentication and *1* negotiation to communicate with certain AP. Because one UT communicates with *m* APs, One UT performs *m* authentications and *m* negotiations in total. From the viewpoint of AP, certain AP performs *n* authentications and *n* negotiations because certain AP communicates with *n* UTs.

In these days, from several reasons (provisioning of authentication information, security of distributed authentication information, and so on), many large enterprise introduce an authentication server (AS) for centralized authentication information management and peer authentication on behalf of UTs/APs. RADIUS server, DIAMETER server and OCSP responder are examples of AS. If AS is introduced, AS authenticates UTs on behalf of all APs. It means that AS performs *nm* authentications in IKE method.

In SIP DIAL-UP method, one UT also establishes an IKE session and an IPsec session with certain AP. It means that one UT also performs *1* authentication and *1* negotiation to communicate with certain AP. In addition, one UT performs *1* pre-authentication before any communications because the UT establishes a SIP session with CSCF. Therefore, one UT performs *1* pre-authentication, *m* authentications and *m* negotiations in total because one UT communicates with *m* APs. From the viewpoint of AP, certain AP performs *1* pre-authentication, *n* authentications and *n* negotiations because certain AP communicates with n UTs. If AS is introduced, AS performs m pre-authentications and nm authentications on behalf of all APs.

In the proposed method, one UT performs *1* pre-authentication when establishing a SIP session with CSCF and performs *no* authentication and *no* negotiation to communicate with certain AP. And one UT may perform *1* pre-authentication when sending a SIP MESSAGE message to sSCP. It means that one UT only performs *1 or 2* pre-authentication in total. From the viewpoint of AP, certain AP also performs only *1* pre-authentication in total even if certain AP communicates with n UTs. If AS is introduced, AS performs m pre-authentications on behalf of all APs.

Therefore, from the viewpoint of processing power, the proposed method takes less processing power than existing two methods as shown in Table 4.3.

(a) A UT using existing two methods is required to perform authentications in proportion to the number of APs (i.e., $O(m)$). On the other hand, a UT using the proposed method is required to only perform a fixed number of pre-authentication (i.e., $O(1)$).

Table 4.3 Comparisons of processing power on the Client-Server model

Method Process		IKE	SIP DIAL-UP	Proposed method
UT	Pre-authentication	***0***	*1*	*1*
	Authentication	*1*	*1*	***0***
	Negotiation of SA	*1*	*1*	***0***
AP	Pre-authentication	***0***	*1*	*1*
	Authentication	n	n	***0***
	Negotiation of SA	n	n	***0***
AS	Pre-authentication	***0***	$n+m$	$n+m$
	Authentication	nm	nm	***0***

(b) A UT using existing two methods is required to perform negotiations in proportion to the number of APs (i.e., *O*(*m*)), but the proposed method is not required to perform *any* negotiation (i.e., *O*(*0*)).

(c) An AP using existing two methods are required to perform authentications in proportion to the number of UTs (i.e., *O*(*n*)), but an AP using the proposed method is required to only perform a fixed number of pre-authentication (i.e., *O*(*1*)),

(d) An AP using existing two methods is required to perform authentications in proportion to the number of UTs (i.e., *O*(*n*)), but an AP using the proposed method is not required to perform *any* negotiation (i.e., *O*(*0*)),

(e) If AS is introduced, an AS using existing two methods are required to perform authentications in proportion to the product of the number of APs and UTs. (i.e., *O*(*mn*)), but an AS using the proposed method is required to perform pre-authentications in proportion to the sum of the number of APs and UTs. (i.e., *O*(*m*+*n*))

(6) Comparison of processing power on the P2P model

In IKE method, one UT establishes an IKE session and an IPsec session with certain UT. It means that one UT performs *1* authentication and *1* negotiation to communicate with certain UT. Because one UT communicates with *n-1* UTs on the P2P model, One UT performs *n-1* authentications and *n-1* negotiations in total. If AS is introduced, AS authenticates other UTs on behalf of all UTs. It means that AS performs *n(n-1)* authentications in IKE method.

In SIP DIAL-UP method, one UT also establishes an IKE session and an IPsec session with certain UT. It means that one UT also performs *1* authentication and *1* negotiation to communicate with certain UT. In addition, one UT performs *1* pre-authentication to establish a SIP session with CSCF. Therefore, one UT performs *1* pre-authentication, *n-1* authentications and *n-1* negotiations in total. If AS is introduced, AS performs *n* pre-authentications and *n(n-1)* authentications on behalf of all APs.

In the proposed method, one UT performs *1* pre-authentication when establishing a SIP session with CSCF and performs *no* authentication and *no* negotiation to communicate with other UTs. And one UT may perform *1* pre-authentication when sending a SIP MESSAGE message to sSCP. It means that one UT only performs *1 or 2* pre-authentication in total. If AS is introduced, AS performs *n* pre-authentications on behalf of all APs.

Therefore, from the viewpoint of processing power, the proposed method takes less processing power than existing two methods as shown in Table 4.4.

(a) A UT using existing two methods is required to perform authentications in proportion to the number of UTs (i.e., $O(n)$). On the other hand, a UT using the proposed method is required to only perform a fixed number of pre-authentication (i.e., $O(1)$).

(b) A UT using existing two methods is required to perform negotiations in proportion to the number of APs (i.e., $O(m)$), but the proposed method is not required to perform *any* negotiation (i.e., $O(0)$).

(c) If AS is introduced, an AS using existing two methods are required to perform authentications in proportion to the square of the number of UTs. (i.e., $O(n^2)$), but an AS using the proposed method is required to perform pre-authentications in proportion to the number of UTs (i.e., $O(n)$).

Table 4.4 Comparisons of processing power on the P2P model

Process	Method	IKE	SIP DIAL-UP	Proposed method
UT	Pre-authentication	***0***	*1*	*1*
	Authentication	*n*	*N*	***0***
	Negotiation of SA	*n*	*N*	***0***
AS	Pre-authentication	***0***	*N*	*n*
	Authentication	n^2	n^2	***0***

(7) Comparison of information management on the Client-server model

In IKE method, one AP requires managing a list of all UT's IDs for authorization. It means that a manager of the enterprise network requires distributing access control information to *m* APs.

In SIP DIAL-UP method, one AP also requires managing a list of all UT's IDs for authorization. It means that a manager of the enterprise network requires distributing access control information to *m* APs.

In the proposed method, access control information is not managed on each AP and is managed on sSCP. It means that a manager of the enterprise network

Therefore, from the viewpoint of information management, existing two methods is required to manage access control information securely on the places in proportion to the number of APs (i.e., $O(m)$) but the proposed method is required to manage access control information securely on sSCP (i.e., $O(1)$).

(8) Comparison of information management on the P2P model

In IKE method, one UT requires managing a list of all other UT's IDs for authorization. It means that a manager of the enterprise network requires distributing access control information to *n* UTs.

In SIP DIAL-UP method, one UT also requires managing a list of all other UT's IDs for authorization. It means that a manager of the enterprise network requires distributing access control information to *n* UTs.

In the proposed method, access control information is not managed on each UT and is managed on sSCP. It means that a manager of the enterprise network

Therefore, from the viewpoint of information management, existing two methods is required to manage access control information securely on the places in proportion to the number of UTs (i.e., $O(n)$) but the proposed method is required to manage access control information securely on sSCP (i.e., $O(1)$).

4.4.3 Considerations for security

(1) Security against the impersonation

In the proposed method, the communication device does not authenticate the communication peer directly. It means that the following risks are considered as a cause of impersonation.

- CSCF misunderstand an attacker with UT, AP or sSCP.
- sSCP misunderstand an attacker with UT or AP.
- An attacker hijacks a SIP session of authenticated user.
- An attacker updates UA information stored in CSCF by unauthorized access.
- An attacker updates SP information stored in sSCP by unauthorized access.

However, CSCF authenticate an entity by TLS with mutual authentication when the entity requests to establish a SIP session and refuse to transmit SIP messages from unauthenticated entity. Therefore, it is difficult that CSCF misunderstand an attacker with UT, AP or sSCP.

And SIP messages received by sSCP are only from authenticated entity because of above mentioned NGN security function. Therefore, sSCP can identify the sender of SIP messages correctly. It means that it is difficult that sSCP misunderstand an attacker with UT or AP.

Against a SIP session hijack, the proposed system can protect a SIP session by TLS. Because all SIP messages are encrypted, it is difficult for an attacker to modify/insert SIP messages. Therefore, it is difficult for an attacker to hijack a SIP session.

To protect the stored information in CSCP or sSCP, access control function provided by most of file systems or database products can be used.

As the result, the proposed system prevents from unauthorized communication by the impersonation of UTs (or APs) and from phishing.

(2) Security of SA distribution

As mentioned above, one of the most serious security threats for sSCP is that an attacker can decrypt an IPsec packet by acquiring an SA illegally.

There are two types of acquisition timing; when the SA is generated on sSCP and when the SA is distributed from sSCP.

To prevent from the former illegal acquisition, sSCP is required not to hold SA (especially secret information like encryption key, message authentication key and so on). Therefore, sSCP abandon an SA as soon as distribution of it even if sSCP generate the SA.

To prevent from the latter illegal acquisition, sSCP is required to protect communication data with UTs and APs. Because communication within NGN is protected by NGN, the problem is how to protect SIP messages between CSCF and UT (or AP) on the Internet. IMS, which is a specification of CSCF, is required to protect communication data between CSCF and the communication terminal by IPsec. Therefore, the confidentiality of SIP messages between CSCF and UT (or AP) is achieved by IPsec.

As the result, it is difficult to acquire SA illegally on the proposed system.

(3) Security of SIP messages

The proposed system protects a SIP message by hop-by-hop security. Namely, SIP messages transmitted between UA in SOHO and CSCF (via the Internet) are protected by TLS with mutual authentication, SIP messages transmitted in NGN are protected by NGN security function, and SIP messages transmitted between sSCP and CSCF are protected by TLS with mutual authentication..

In addition, once sSCP authenticates UAs, sSCP considers that the sender of SIP message, which sSCP receives, is already authenticated because NBE prevent from transmitting SIP messages from unauthorized entity.

As the result, the proposed system protects all of SIP messages.

(4) Security against the man-in-the-middle attack

Man-in-the-middle attack is an attack that an illegal intermediate makes independent connections and relays data with modifications.

There are two types of man-in-the-middle attacks on the proposed system; man-in-the-middle attack for a SIP session and man-in-the-middle attack for an application data

To prevent from the man-in-the-middle attack for a SIP session, the proposed system protects a SIP session by hop-by-hop security as mentioned above.

To prevent from the man-in-the-middle attack for an application data, the proposed system protects by end-to-end security. Namely, an application data exchanged between two communication devices are encrypted by IPsec. And also, an SA is distributed securely via secure SIP session. Therefore, the man-in-the-middle attack for an application data is difficult.

4.4.4 Considerations for implementation

(1) Separation of IPsec and IKE

In these days, many of operating systems (OSs) provide IPsec communication function. Therefore, to implement the proposed method to communication devices, it will be required to install software modules, which implement SPR function and SSC function, instead of IKE function.

However, there are some OSs on which IPsec communication function is integrated with IKE function. For that case, it is also required to implement IPsec function and SADB as a virtual network driver and to disable the IPsec function provided by OS.

(2) Underlying protocol of SIP

Because some SIP messages are digitally signed and/or contains SP or SA, the size of message is bigger than 1500 octets (the size of one IP packet). Therefore, it is recommended not to use UDP, but to use TCP as the transport layer protocol.

(3) Load balancing of sSCP

sSCP may be a single point of failure in the proposed method. Especially, there is a possibility to exceed the maximum processing ability of sSCP if numerous numbers of UTs and/or APs perform the authentication phase and/or the key distribution phase at same time. To avoid it, it is required to consider the load balancing of sSCP.

4.5 Supporting TLS

The secure communication provider service is also able to apply to the Transport Layer Security (TLS) communication instead of IPsec communication.

Generally, the TLS handshake process takes a lot of costs to verify the authenticity of the communication peer because the peer authentication of TLS is based on PKI. And also, key calculation in negotiation of security association is a costly process because public key encryption/decryption or DH key exchange methods are used.

The resuming session can improve the performance of TLS session establishment ([52][53]). The resuming session is known as "TLS session resume" shown in Figure 4.8.

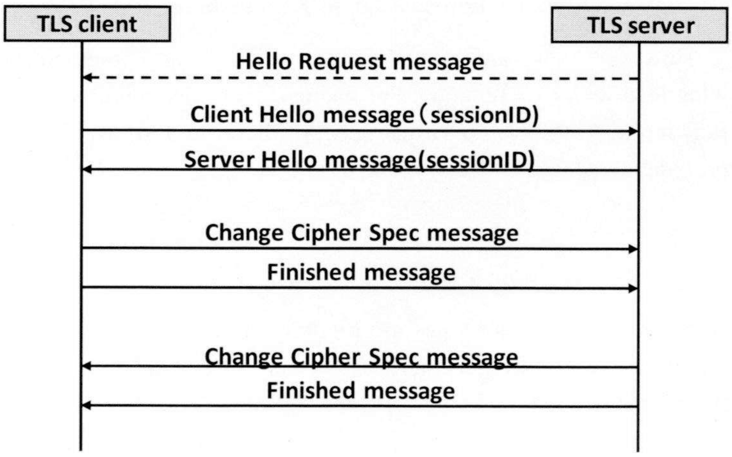

TLS client
TLS server
Hello Request message
Client Hello message (sessionID)
Server Hello message(sessionID)
Change Cipher Spec message
Finished message
Change Cipher Spec message
Finished message

Figure 4.8 TLS session resume

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <xsd:element name="TLS">
  <xsd:complexType>
   <xsd:sequence>
    <xsd:element name="CipherSuite" type="xsd:string"
                                   maxOccurs="unbounded" />
   </xsd:sequence>
  </xsd:complexType>
 </xsd:element>
</xsd:schema>
```

Figure 4.9 XML schema of SP for TLS

```
<TLS>
  <CipherSuite>TLS_RSA_WITH_AES_128_CBC_SHA</CipherSuite>
  <CipherSuite>TLS_RSA_WITH_3DES_EDE_CBC_SHA</CipherSuite>
</TLS>
```

Figure 4.10 Example of SP for TLS

To support TLS communication, following modifications are required.

MESSAGE message is sent from TLS client (TLS server) to SIP server. The body of this message carries a security policy (i.e., a list of available cipher suites). Figure 4.9shows the XML schema of security policy. And Figure 4.10 shows an example of security policy.

1. Registering available cipher suite as SP

   To support TLS communication, it is required to register available cipher suites for TLS as SP in the authentication phase. Figure 4.9 is a XML schema representation for SP. (Figure 4.10 is an example of SP.)

   The security policy in Figure 8 states that TLS client (or TLS server) has two available cipher suites: one is the cipher suite "TLS_RSA_WITH_AES_128_CBC_SHA", which uses the RSA public key encryption algorithm, the AES common key encryption algorithm with 128 bits key in CBC mode and the SHA-1 hash algorithm for peer authentication, data encryption and HMAC respectively. And another is the cipher suite "TLS_RSA_WITH_3DES_EDE_CBC_SHA", which uses the RSA public key encryption algorithm, the triple DES common key encryption algorithm in EDE and CBC mode and the SHA-1 hash algorithm for peer authentication, data encryption and HMAC respectively.

2. Distributing selected cipher suite and master secret as SA

   To support TLS communication, SA in the key distribution phase is required to contain the information enough to TLS session resume. That is the session ID, the selected cipher suite, the master secret value, the life time of this security association. Figure 4.11 is a XML schema representation for SA. (Figure 4.12 is an example of SA.)

   It notes that a security association compliant with Figure 10 might contain a compression method a TLS client information, which is put between "<ClientInfo>" and "</ClientInfo>", and a TLS server information, which is put between "<ServerInfo>" and "</ServerInfo>", for this session. The TLS client information and the TLS server information can include a random value for this TLS session, a pair of IP address and port number, hostname, URI or Public Key Certificate of TLS client or TLS server

3. SA confirmation in the key distribution phase

   After distributing SA, the SA confirmation is performed immediately in the key distribution phase. TLS server and TLS client confirm if they share a same security association or not by the TLS session resume procedure. If it finishes successfully, TLS server and TLS client start the application data transmission through the established TLS session.

   During the SA confirmation, the following TLS Handshake messages are exchanged between TLS client and TLS server.

(1) TLS client refers the security association shared at the key distribution phase and sends TLS server a ClientHello message with the session ID of security association.

(2) When TLS server receives the ClientHello message, TLS server returns a ServerHello message with same session ID to TLS client if TLS server can find the security association, which corresponds to the session ID indicated by the ClientHello message.

(3) Then, TLS client sends a ChangeCipherSpec message and a Finished message.

(4) TLS server sends a ChangeCipherSpec message and a Finished message if TLS server can decrypt the Finished message received from TLS client.

```
<?xml version="1.0" encoding="ISO-2022-JP"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <xsd:element name="TLS">
  <xsd:complexType>
   <xsd:sequence>
    <xsd:element name="SessionID" type="xsd:string" />
    <xsd:element name="CipherSuite" type="xsd:string" />
    <xsd:element name="CompressionMethod" type="xsd:string"
                              minOccurs="0" />
    <xsd:element name="MasterSecuret" type="xsd:string" />
    <xsd:element name="Lifetime" type="xsd:int" />
    <xsd:element name="ClientInfo" minOccurs="0">
     <xsd:complexType>
      <xsd:sequence>
       <xsd:element name="random" type="xsd:string"
                         minOccurs="0" />
       <xsd:element name="Certificate" type="xsd:base64Binary"
                              minOccurs="0" />
       <xsd:element name="ipAddr" type="xsd:string"
                         minOccurs="0" />
       <xsd:element name="portNum" type="xsd:unsignedShort"
                          minOccurs="0" />
       <xsd:element name="hostname" type="xsd:string"
                          minOccurs="0" />
       <xsd:element name="URI" type="xsd:string"
                       minOccurs="0" />
      </xsd:sequence>
     </xsd:complexType>
    </xsd:element>
    <xsd:element name="ServerInfo" minOccurs="0">
     <xsd:complexType>
      <xsd:sequence>
       <xsd:element name="random" type="xsd:string"
                         minOccurs="0" />
       <xsd:element name="Certificate" type="xsd:base64Binary"
                              minOccurs="0" />
       <xsd:element name="ipAddr" type="xsd:string"
                         minOccurs="0" />
       <xsd:element name="portNum" type="xsd:unsignedShort"
                          minOccurs="0" />
       <xsd:element name="hostname" type="xsd:string"
                          minOccurs="0" />
       <xsd:element name="URI" type="xsd:string"
                       minOccurs="0" />
      </xsd:sequence>
     </xsd:complexType>
    </xsd:element>
   </xsd:sequence>
  </xsd:complexType>
 </xsd:element>
</xsd:schema>
```

Figure 4.11 XML schema for SA for TLS

```
<TLS>
  <SessionID>114454-132344-1475452@bar.hitachi.com</SessionID>
  <CipherSuite>TLS_RSA_WITH_AES_128_CBC_SHA</CipherSuite>
  <MasterSecret>MK8wuzC5jiMQEwTVH ••• MQAwADgA8w</MasterSecret>
  <Lifetime>36000</Lifetime>
</TLS>
```

Figure 4.12 Example of SA for TLS

(5) TLS client and TLS server transmit application data if TLS client can decrypt the Finished message received from TLS server.

4.6 Conclusion

In this section, the secure communication provider service is described. This service is to protect the confidentiality and integrity of communications crossing over NGN and the Internet by setting up and controlling the IPsec session in cooperation with NGN's call session control function (CSCF). In this service, the secure session control provider (sSCP) server and CSCF authenticate its user as Trusted Third Party on behalf of service providers. In addition, this service provides the ability of fast session establishment because sSCP distributes a security association for IPsec session between the user and service provider via the SIP session protected by NGN.

In general, the proposed method is superior to existing two methods (IKE and SIP DIAL-UP) if the number of communication device is more and/or these devices perform mesh communications.

The performance evaluation of the secure session provider service, the policy update function in cooperation with user management function of NGN and the support of IP multicast protocol are further study.

Chapter 5
TLS implementation for cross certification model

5.1 Introduction

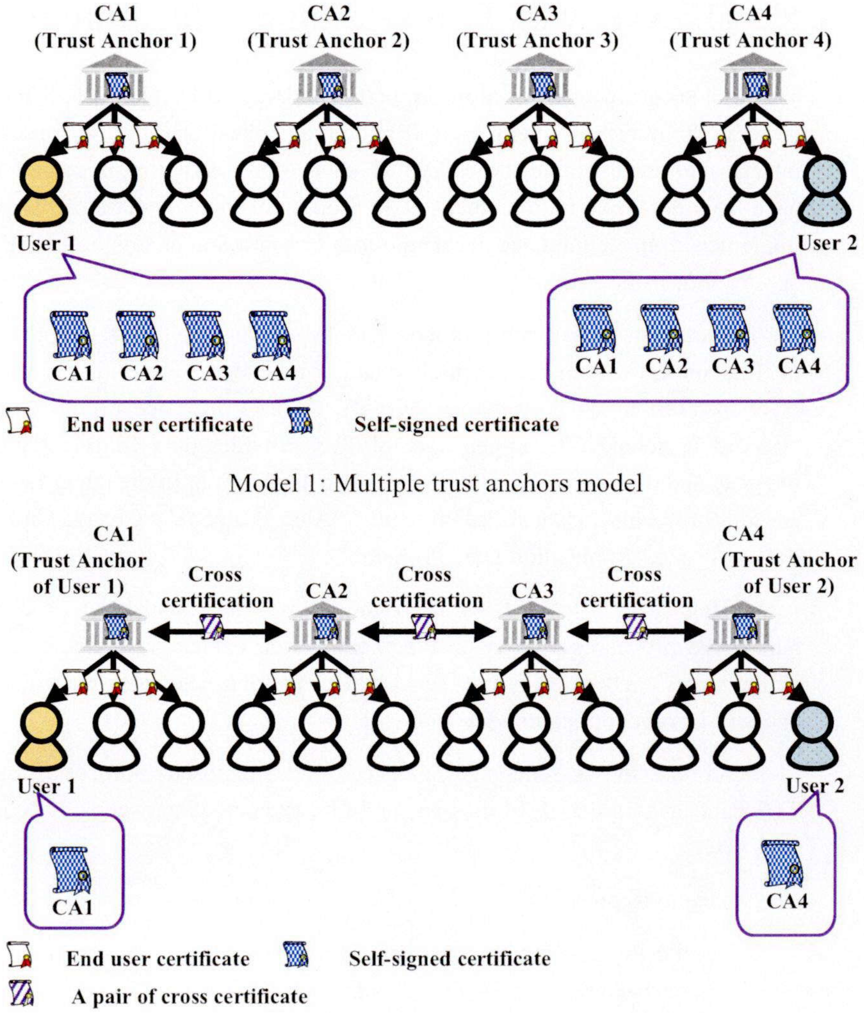
This chapter proposes a TLS implementation method for the cross certification model to allow to manage a huge number of CAs.

For the enterprise network over NGN, the protection of signaling messages (i.e. SIP messages) is very important. The specifications of SIP recommend protecting SIP messages by using TLS based on PKI for entity authentication or data origin authentication. Terminals in the enterprise network and NGN will be required to manage huge number of certification authorities (CAs) if enterprises and NGN service providers will operate their own PKI respectively.

There are two models for the interconnection with multiple PKI environments; "multiple trust anchors environment" (Mobile 1 of Figure 5.1) and "cross certification environment" (Mobile 2 of Figure 5.1). If there are many CAs, the multiple trust anchors environment causes the problem that the verifier has to maintain huge number of CA certificates. Therefore, the cross certification environment is useful.

However, TLS will not be able to work efficiently on the cross-certification model because TLS is designed for the multiple trust anchors model.

This thesis proposes a profile of TLS handshake protocol for the cross certification environment and a TLS implementation method with a delegated certification path discovery and validation server for the cross certification model efficiently.



Model 1: Multiple trust anchors model

Model 2: Cross-certification model

Figure 5.1 Two models of multiple PKI environment

5.2 Overview of TLS session establishment

TLS is a secure communication protocol standardized by IETF. TLS located in between the transport layer (i.e., TCP) and the application layer. TLS uses socket interface to send/receive the encrypted data. Many activities in regard to TLS have been done like improvement of TLS protocol to adopt certain environment, implementation technique to accelerate data transmission or session establishment, applications using TLS and so on. ([54][55])

TLS consists of five sub-protocols and they forms two layers. The lower layer of TLS provides data confidentiality and data integrity functions to the upper layer of TLS. In the lower layer of TLS, there is only one protocol called as "Record Protocol." The upper layer of TLS provides data confidentiality, data integrity and peer authentication functions to application. In the upper layer, there are four protocols, "Handshake Protocol," "Alert Protocol," "Change Cipher Spec Protocol" and "Application Data Protocol."

" Handshake Protocol" is used to perform the handshake process. The handshake process performs peer authentication, exchanges a set of available cipher suites each other, selects one cipher suite and calculates a session key to encrypt/decrypt application data.

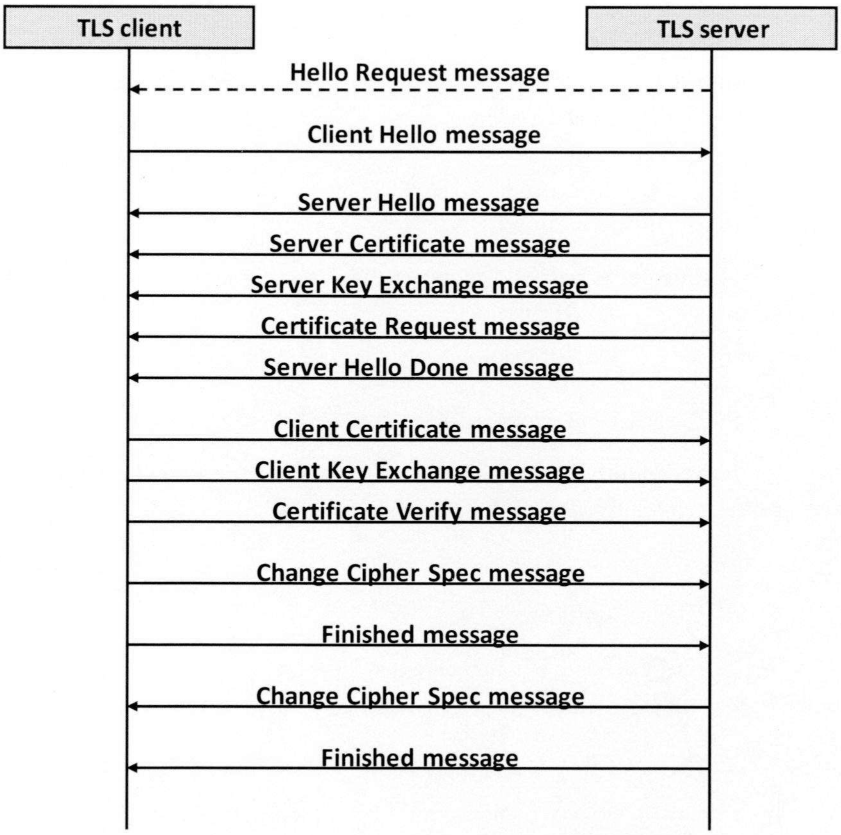
Figure 5.2 shows a message sequence of TLS handshake process. During the TLS handshake process, 14 messages will be exchanged between TLS client and TLS server.

(1) Hello Request

Hello Request is a message that TLS server requests to start the new TLS handshake process for TLS client.

(2) Client Hello

Client Hello is a message that TLS client informs TLS server about its protocol version, session ID and a list of algorithms.



TLS client
TLS server
Hello Request message
Client Hello message
Server Hello message
Server Certificate message
Server Key Exchange message
Certificate Request message
Server Hello Done message
Client Certificate message
Client Key Exchange message
Certificate Verify message
Change Cipher Spec message
Finished message
Change Cipher Spec message
Finished message

Figure 5.2 Message sequence of TLS handshake process

(3) Server Hello

Server Hello is a message that TLS server responds to Client Hello and contains selected algorithms.

(4) Server Certificate

Server Certificate is a massage that TLS server sends the certificate chain from TLS server certificate to its trust anchor certificate.

(5) Server Key Exchange

Server Key Exchange is a message that TLS server sends the information to exchange the premaster secret.

(6) Certificate Request

Certificate request is a message that TLS server requests to send TLS client certificate.

(7) Server Hello Done

Server Hello Done is a message that TLS server indicates the end of the server hello and associated messages.

(8) Client Certificate

Client Certificate is a message that TLS client sends the certificate chain from TLS client certificate to its trust anchor certificate to TLS server.

(9) Client Key Exchange

Client Key Exchange is a message that TLS client sends the premaster secret to TLS server.

(10) Certificate Verify

Certificate Verify is a message that TLS client sends the data to verify TLS client certificate.

(11) Change Cipher Spec

Change Cipher Spec is a message to notify that subsequent records will be protected under the newly negotiated algorithms and keys.

(12) Finished

Finished message is a message to verify that the key exchange and authentication processes were successful.

5.2.1 TLS extensions in related to certificate validation

IETF standardized some extensions of TLS specification as [41] and [42]. And there is one extension in related to certificate validation called "Certificate Status Request" extension in [43].

The Certificate Status Request extension is for TLS client to wish to use OCSP ([46]) to verify the TLS server certificate in order to avoid transmission of CRLs. The TLS client provides a list of OCSP responders, which the TLS client trusts, in Client Hello. And the TLS server returns an OCSP response to the TLS client along with TLS server certificate as a "Certificate Status" message immediately after Server Certificate.

5.3 Features of the cross-certification model

At first, it should describe some features of the cross-certification model, which are related to the TLS session establishment.

Feature 1: Less computing power and memory

In the recent, some terminals in SOHO will have less computing power and memory than current PCs.

The cross-certification model can save memory size to hold the trust anchor certificate.

However, it will take more computing power than the multiple trust anchors model as mentioned below.

Feature 2: Equality of terminals

Terminals performing P2P communication will act as both "clients" and "servers" to the other terminals on the network.

Feature 3: Different trust anchors

In the cross-certification model, a major difference from the multiple trust anchor model is that a certificate user and a relying party will have different trust anchor.

Therefore, to verify a certification path from the certificate user to the relying party's trust anchor, the relying party should discover a certification path from the certificate user's trust anchor to the relying party's trust anchor.

Feature 4: Complex certification path discovery

In the cross-certification model, the process of certification path discovery is more complex than the multiple trust anchor model.

In the multiple trust anchor model, there is a hint on current certificate to discover next certificate. Thus the issuer name of the current certificate is the subject name of the next certificate. For example, in Figure 1, the issuer name of user 1's certificate is the name of CA1. In other words, if the issuer name of the target certificate is unknown name, the target certificate is unreliable.

On the other hand, in the cross-certification model, there is no hint on the cross certificate. For example, in Model 2 of Figure 5.1, there is no information that CS3 issued the cross certificate issued to CA2, on the cross certificate issued from CA2 to CA1. In other words, even if the issuer name of the target certificate is an unknown name, the target certificate may be reliable. Therefore, to discover the certification path in the cross-certification model, it requires implementing the path discovery algorithm between two vertexes in the directed graph, and constructing the certificate chain by walking around LDAP server.

Feature 5: Full certification path validation

In the cross-certification model, intermediate certificates in the certification path might not be trustable. Therefore the relying party requires verifying the certification path by retrieving CRL/ARL from LDAP server or asking OCSP responder for all of certificates in the certification path.

As described in Section 5.2.1, the Certificate Status Request extension can request to send the current status of TLS server certificate.

However, in the cross-certification model, even if the TLS server certificate is valid, it does not mean that there is a valid certification path from the TLS server to the TLS client's trust anchor because one of CA certificate in the certification path may be revoked.

For example, in Model 2 of Figure 5.1, if the cross certificate issued from CA3 to CA2 is revoked, there is no certification path from user 1 to CA4.

Therefore, the Certificate Status Request extension may be useless in the cross-certification model.

5.4 Requirements of TLS implementation for the cross-certification model

This section describes requirements of TLS implementation that adapts for the cross-certification model. There are the following four requirements.

Requirement 1: Certification path discovery function on the TLS client

In the cross-certification model, the TLS server's trust anchor might not be the TLS client's trust anchor. Therefore, TLS client requires having the certification path discovery function to verify the TLS server certificate.

Requirement 2: Certification path validation function on the TLS client

TLS client also requires having the certification path validation function.

Requirement 3: Certification path validation function on the TLS server

Because TLS client can know the TLS server's trust anchor from the certificate chain contained in Server Certificate, TLS client can construct the certificate chain from TLS client certificate to TLS server's trust anchor certificate. Thus the TLS server will not require having the certification path discovery function. However, TLS server requires having the certification path validation function to verify the certificate chain of TLS client.

Requirement 4: Less computing power

Some terminals in SOHO will have less computing power than the current PCs.

Therefore, the TLS implementation should take computing power as few as possible.

5.5 TLS implementation method for the cross-certification model

This section proposes two implementation methods of the TLS client and the TLS server to support the cross-certification model.

5.5.1 Design principals of proposed TLS implementation

To meet requirements in Section 5.4, we propose the following two design principles for the TLS implementation to adapt for the cross-certification model.

(a) Equal functionality of TLS client and TLS server

From the feature 5 in Section 5.3, terminals will require to act as both TLS client and TLS server if terminals perform P2P communication.

In other words, the TLS client and the TLS server can have the same functionality.

In Section 5.4, TLS server requires implementing the certification path validation function only (Requirement 3). However, the TLS client requires implementing the certification path discovery function and the certification path validation function (Requirement 1 and 2) .

Therefore, the TLS server also has the certification path discovery function.

As the result that both the TLS client and the TLS server have the certification path discovery function and the certification path validation function, the size of exchanged messages between the TLS client and the TLS server can be reduced. Namely, the TLS server and the TLS client do not send the certificate chain, but send its own certificate only because the certification path discovery function retrieves one or more certification path from the target certificate and the trust anchor certificate (if it exists).

(b) Delegation of Path Discovery and Validation

As mentioned above, both the TLS client and the TLS server have the certification path discovery function and the certification path validation function.

However, these processes take much computing cost.

To reduce these computing costs, both the TLS client and the TLS server implement the function that is to ask their trust DPD&DPV ([47]) server to discover or validate a certification path instead of implementing the certification path discovery function and the certification path validation function. (The DPD&DPV server is a server to have a certification path discovery function and a certification path validation function. Currently, there are some standardized protocols between a DPD&DPV server and its client (like SCVP ([48]), X-KISS ([49]) and CVS protocol ([50])) and some implementations.)

As the result, both the TLS client and the TLS server can reduce the computing cost of certification path discovery and certification path validation.

5.5.2 Overview of proposed implementation method

Figure 5.3 shows functional units of a TLS server and a TLS client in the proposed implementation method.

In this figure, RPF is a functional unit to handle the TLS Record protocol, CCSPF is a functional unit to handle the TLS Change Cipher Spec Protocol, APF is a functional unit to handle the TLS Alert Protocol, ADPF is a functional unit to handle the TLS Application Data Protocol, HPF is a functional unit to handle the TLS Handshake Protocol extended to adapt for the cross-certification model, DPD&DPVCF is a functional unit to ask a DPD&DPV server to discover and/or validate a certification path of communication peer's certificate and DPD&DPVSF is a functional unit to discover and/or validate a certification path.

In the above, RPF, CCSPF, APF and ADPF do not have any modification to adapt for the cross-certification model.

(a) TLS Handshake Protocol Processing Function Unit (HPF)

HPF handles the TLS Handshake Protocol, which is extended for cross-certification model. Namely, when HPF receives the communication peer's certificate(s), HPF asks DPD&DPVCF if the certificate is valid or not, and continues the handshake process if and only if DPD&DPVCF returns that the certificate is valid.

(b) Delegated Path Discovery and Validation Server Function Unit (DPD&DPVSF)

DPD&DPVSF constructs the certification path between given two certificates by walking around LDAP server and validates the certification path by retrieving CRL/ARL from LDAP server or asking OCSP responder for all of certificates in the certification path.

And DPD&DPVSF returns the result of certification path validation. Namely, if DPD&DPVSF can construct the certification path and it is valid, DPD&DPVSF returns a success message. Otherwise, DPD&DPVSF returns a failure message.

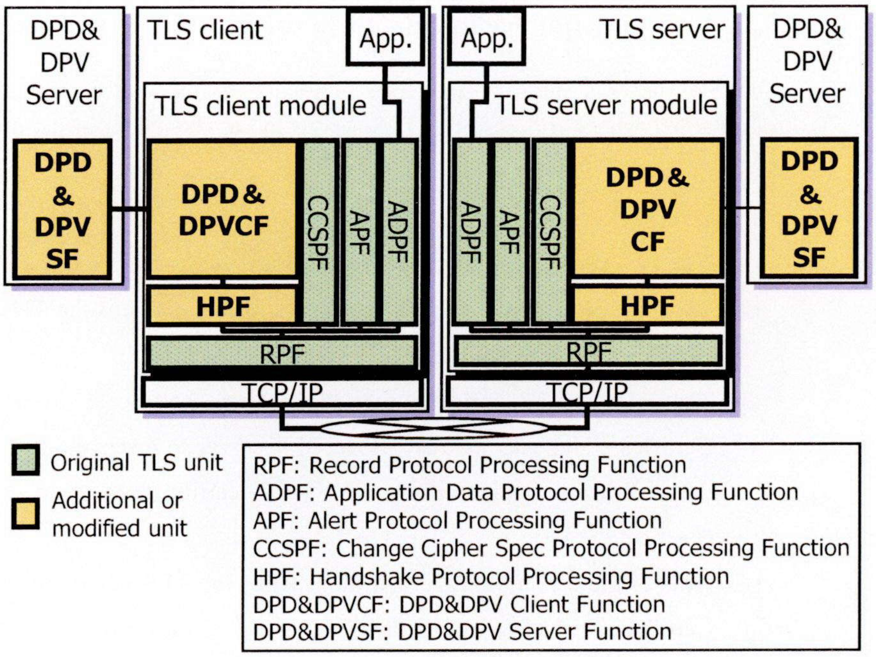

DPD& DPV Server
DPD & DPV SF
TLS client
App.
TLS client module
DPD & DPVCF
CCSPF
APF
ADPF
HPF
RPF
TCP/IP
App.
TLS server
TLS server module
ADPF
APF
CCSPF
DPD & DPV CF
HPF
RPF
TCP/IP
DPD& DPV Server
DPD & DPV SF
🟩 Original TLS unit
🟨 Additional or modified unit
RPF: Record Protocol Processing Function
ADPF: Application Data Protocol Processing Function
APF: Alert Protocol Processing Function
CCSPF: Change Cipher Spec Protocol Processing Function
HPF: Handshake Protocol Processing Function
DPD&DPVCF: DPD&DPV Client Function
DPD&DPVSF: DPD&DPV Server Function

Figure 5.3 Overview of proposed implementation method

(c) Delegated Path Discovery and Validation Client Function Unit (DPD&DPVCF)

DPD&DPVCF realizes the certification path discovery function and the certification path validation function by asking DPD&DPVSF.

DPD&DPVCF keeps its trust anchor certificate.

When DPD&DPVCF is asked the validity of certificate, DPD&DPVCF asks DPD&DPVSF to discover and validate the certification path from the target certificate to its trust anchor certificate. And if DPD&PDVCF receives a success message from DPD&DPVSF, DPD&DPVCF returns that the certificate is valid. Otherwise, DPD&DPVCF returns that the certificate is invalid.

5.5.3 Message sequence in related to the certificate validation

Figure 5.4 shows a message sequence in related to the certificate validation during the handshake process on a TLS server and a TLS client in the proposed implementation method.

In Figure 5.4,

(1) The TLS client sends Client Hello to the TLS server, at first.

(2) The TLS server sends Server Certificate, which contains the TLS server certificate only, back to the TLS client.

(3) The TLS client retrieves the TLS server certificate from Server Certificate and asks its trust DPD&DPV server to discover and validate the certification path from the TLS server certificate to its trust anchor certificate.

(4) (And if there is a valid certification path,) the TLS client sends Client Certificate, which contains the TLS client certificate only, back to the TLS server.

(5) The TLS server retrieves the TLS server certificate from Client Certificate and asks its trust DPD&DPV server to discover and validate the certification path form the TLS client certificate to its trust anchor certificate.

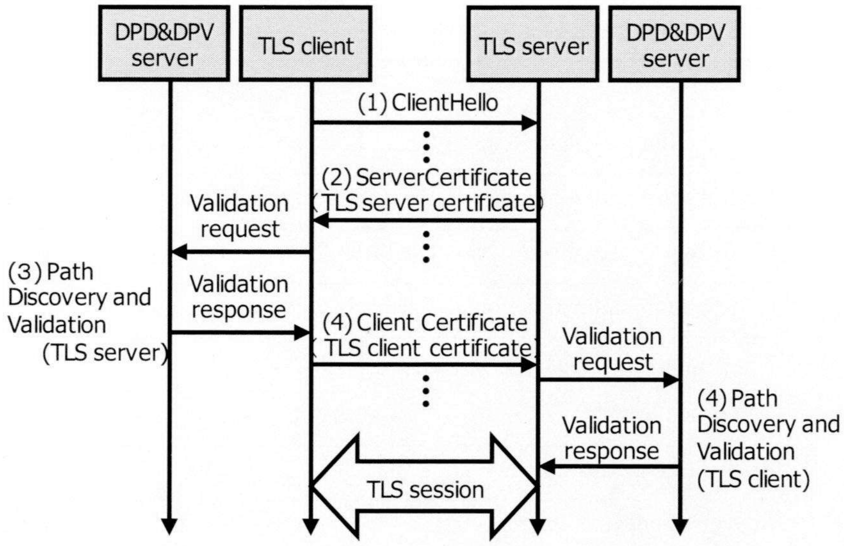

DPD&DPV server
TLS client
TLS server
DPD&DPV server
(1) ClientHello
(2) ServerCertificate (TLS server certificate)
Validation request
(3) Path Discovery and Validation (TLS server)
Validation response
(4) Client Certificate (TLS client certificate)
Validation request
(4) Path Discovery and Validation (TLS client)
Validation response
TLS session

Figure 5.4 Message sequence in related to the certificate validation

5.6 Evaluation

5.6.1 Comparison between TLS specification and proposed method

This subsection comprises between the implementation, which complies with the current TLS specification, and the proposed implementation method from the viewpoint of the cost of certification path discovery/validation, the size of exchanged message during the handshake process. (Table 5.1)

(a) Comparison of the cost of certification path discovery/validation

As mentioned in Section 5.4, TLS server requires implementing the certification path validation function only (Requirement 3). And, the TLS client requires implementing the certification path discovery function and the certification path validation function (Requirement 1 and 2).

Table 5.1 Comparison of two implementation methods

		TLS Spec.	Proposal
Certification path discovery on the TLS Server		Not required	Required
Certification path validation on the TLS Server		Required	Required
Certification path discovery on the TLS Client		Required	Required
Certification path validation on the TLS Client		Required	Required
# of query to the DPD&DPV server	TLS server	1	1
	TLS client	2	1
Size of Server Certificate		Mid	Small
Size of Client Certificate		Large	Small

On the other hand, the proposed implementation method requires that both the TLS server and the TLS client have the certification path discovery function and the certification path validation function.

However, both can reduce the processing cost on the TLS client and the TLS server because DPD&DPV servers carry out the certification path discovery and validation on behalf of the TLS client and the TLS server.

If DPD&DPV servers are used, the processing cost of the certification path discovery and validation on the TLS client or the TLS server equals a number of queries to the DPD&DPV server.

The implementation, which complies with current TLS specification, requires three queries. (First one is a validation request of the TLS server certificate from the TLS client. Second one is a discovery request of a certification path between the TLS client certificate and the TLS server's trust anchor certificate from the TLS client. And third one is a validation request of the certification path from the TLS server)

On the other hand, the proposed implementation requires only two queries.

(b) Comparison of the size of exchanged message during the handshake process

Because certificates are much larger than other potion of messages exchanged during the handshake process, the comparison of the message size almost equals the comparison of a number of certificates exchanged during the handshake process.

The proposed implementation exchanges only the TLS server certificate and the TLS client certificate. On the other hand, the implementation complying with the current TLS specification exchanges certificate chains from the TLS server certificate to the TLS server's trust anchor certificate and from the TLS client certificate to the TLS server's trust anchor certificate. Therefore, the proposed implementation exchanged fewer certificates than the implementation complying with TLS specification.

5.6.2 Prototype implementation

We made a prototype system that implement the proposed design and evaluated it on the PKI shown on Model 2 of Figure 5.1.

Figure 5.5 shows the overview of the prototype system and Table 5.2 shows certificate profiles of the prototype system.

To establish one TLS session, the prototype system exchanges 2028 octets messages between the TLS client and the TLS server in total. (The TLS server sends 875 octets messages and the TLS client sends 1153 octets messages. (Table 5.3)

And, it takes 502 msec to establish one TLS session. (Table 5.4) It includes the processing time on the TLS client (129 msec), the processing time on the TLS server (207 msec). It notes that the processing time includes the response time of DPD&DPV server (45msec).

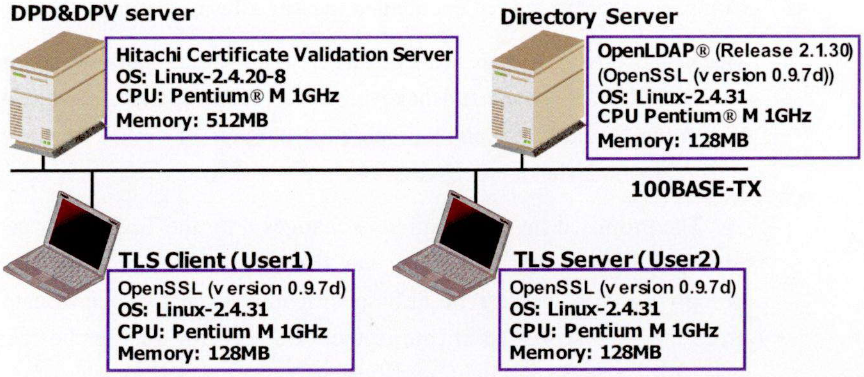


Figure 5.5 Overview of the prototype system

Table 5.2 Certificate profiles

	CA Cert.	**Cross Cert.**	**TLS server**	**TLS client**
Public key algorithm (key length)	RSA (2048bits)	RSA (2048bits)	RSA (1024bits)	RSA (1024bits)
Hash algorithm	SHA-1	SHA-1	SHA-1	SHA-1
Size of cert.	869 bytes	817 bytes	729 bytes	729 bytes

- Pentium is trademark or registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.
- Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Table 5.3 Message size during the handshake process

	Size
Size of Server Certificate	739 octets
Size of Client Certificate	739 octets
Size of messages sent from TLS server	875 octets
Size of messages sent from TLS client	1153 octets
In total	2028 octets

Table 5.4 Elapsed time during the handshake process

	Time
The processing time on the TLS client	129 msec
The processing time on the TLS server	207 msec
The time for TLS session establishment	502 msec

5.7 Conclusion

PKI environment called as "multiple trust anchors environment" is widely used. However it causes the problem that the verifier has to maintain huge number of CA certificates because the increase of terminals connected to the network brings the increase of CAs. However, some of terminals in SOHO will not have enough memory to hold such huge number of CA certificates. Therefore, another PKI environment, "cross-certification environment", is useful.

TLS is a secure communication protocol that uses PKI. But, because current TLS is designed for the multiple trust anchors model, TLS cannot work efficiently on the cross certification model.

This thesis shows features of the cross certification model and derives requirements to meet them at first.

And then, this thesis proposes TLS implementation method to support the cross certification model efficiently. This implementation method provides equal functionality between the TLS server and the TLS client in related to certificate validation and introduces DPD&DPV servers to perform certification path discovery and validation on behalf of the TLS server and the TLS client.

The TLS client and the TLS server using proposed method do not have the certification path discovery and validation function. (For example, LDAP access function to retrieve certificates and CRLs/ALRs and OCSP access function to retrieve certificate status)

The proposed method also reduces the size of exchanged messages between the TLS client and the TLS server during the handshake process.

Therefore, the proposed method is suitable for implementing TLS in the terminals that do not have enough computing power and memory.

Chapter 6
Seamless Object Authentication

The Seamless Object Authentication (SOA) approach described in this section is a solution against the problem that access from a signed malicious code cannot be prevented by security functions of NGN once the home gateway accepts a signed malicious code.

Hereinafter, the Section 6.1 describes above security problem more in detail. And the Section 6.2 describes the Seamless Object Authentication (SOA) approach, the Section 6.3 describes application of SOA approach to existing distributed object system and the Section 6.4 evaluates a Java system with SOA approach with a traditional Java RMI system.

6.1 Security problems of mobile code in the networked system

This section describes the security problem that access from a signed malicious code cannot be prevented by security functions of NGN once the home gateway accepts a signed malicious code more in detail.

As mentioned above, NGN and mobile code system has various security functions. However, careless of developer, distributer and/or user of mobile code will cause security problems from which security functions do not prevent.

One of the security problems from careless of developer is vulnerabilities of a mobile code. (ex. vulnerability against buffer overflow run)

One of the security problems from careless of distributer or user is that a user trusts a signed mobile code even if it is not trusted by a system manager, and allows to run it because of misunderstanding. (Figure 6.1)

- Java is registered trademarks of Oracle and/or its affiliates.

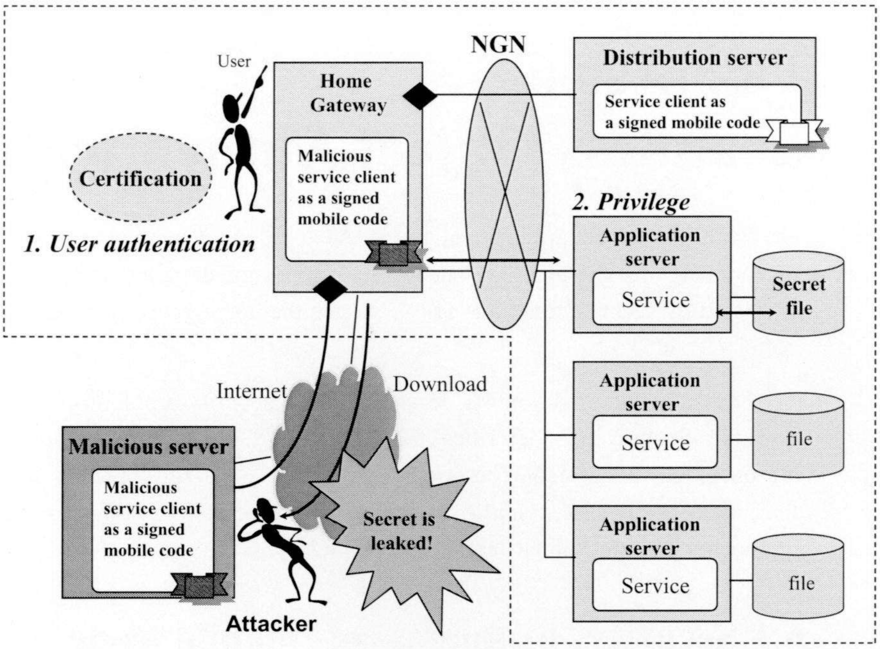


Figure 6.1 Example of security problem from careless of user

The reason of this problem is that an attacker is able to act as a privileged user because of following characteristics of signed mobile code and access control function:

- Signed mobile code is able to access any kinds of resources
- NGN and other distributed system make an access decision based on its "privileged user"

In Figure 6.1, the application server provides a file transferring service to a home gateway. This file transferring service allows a user to retrieve a file if the user has a privilege to access the file.

In the normal case, when the user downloads a service client (a signed mobile code) from a distribution server to the home gateway, the service client is executed automatically, and it accesses the file transferring service.

On the other hand, an attacker develops a malicious service client, which downloads a secret file from an application server and sends it to the attacker, as a mobile code is signed by the attacker itself.

If the user downloads the malicious service client and the user trusts the attacker from careless, the service client is executed with the user's privilege and accesses to the application server. The application server checks if the user has an enough privilege to access the secret file and sends the secret file to the service client. The service client transfers the secret file to the attacker. As the result, the attacker is able to retrieve the secret file, which is not allowed for the attacker to access.

6.2 Overview of seamless object authentication (SOA) approach

This section describes the proposed solution called as "Seamless Object Authentication (SOA)" against the problem as mentioned above.

The basic idea of SOA approach is not only that the user checks if the service client is authorized one or not, but also that the application server checks if the service client is authorized or not.

For this purpose, the user terminal retrieves a digital signature of the service client and send it with the service request message and the application server checks not only if the user has the privilege to access, but also if the digital signature is made by a trusted developer (distributer).

In the system implemented the SOA approach, a mobile code execution platform of the user terminal is required to have a function to retrieve a digital signature of the mobile code and send it to a communication peer when the mobile code tries to communicate with the communication peer. And also, the application server is required to have a function to checks if the digital signature received from the user terminal is made by a trusted developer (distributer) or not during making the access decision.

As the result, even if the malicious service client is executed by the careless of the user, the application server is able to refuse the service request from the malicious service client.

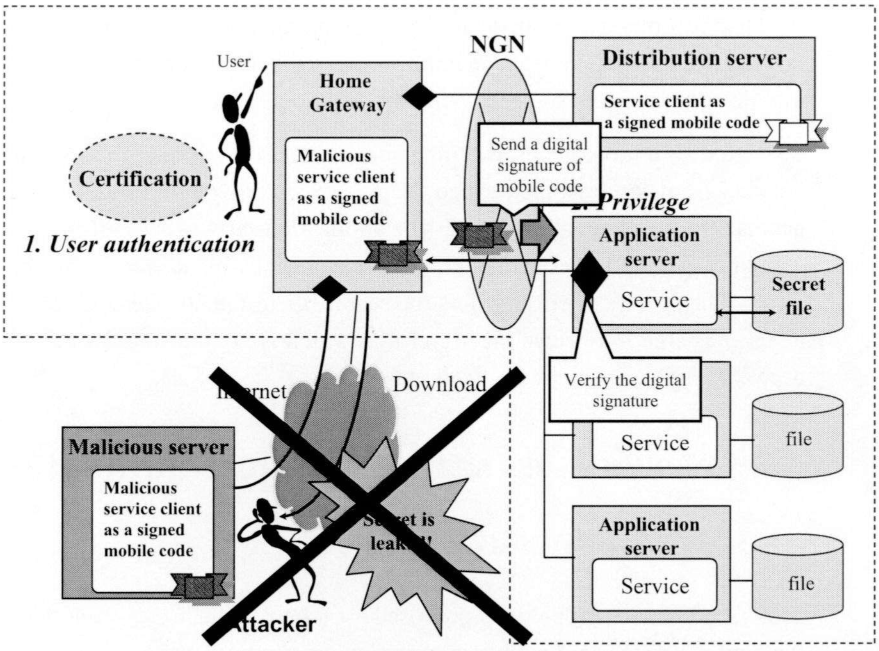


Figure 6.2 SOA approach

6.3 Java Implementation of SOA approach

This section describes one of the implementations of SOA approach, which is used Java Remote Method Invocation (Java RMI).

Hereinafter, the Java system which implements the SOA approach is called "Java system with SOA".

6.3.1 Java RMI system

At first, a system using existing Java RMI technology (called as "Java RMI system") is described briefly.

The Java RMI system consists of a distribution server to store a client object as a signed Java Applet and distributes it, an application server to execute a target service object and a user terminal to execute the client object. The target service object provides certain service to the client object remotely.

The client object sends a service request message to the target service object and receives a service from the target service object.

Object Request Broker (ORB) is a middleware running on the user terminal and the application server and has following two functions:

- Notifying the location where the specified target service object is executed. (called as "bind" process)

  It notes that this function performs to launch the target service object if the target service object is not executed.
- Intermediating a communication between the client object and the target service object.

Once the client object is downloaded from the distribution server, following processes shown in Figure 6.3 are performed:

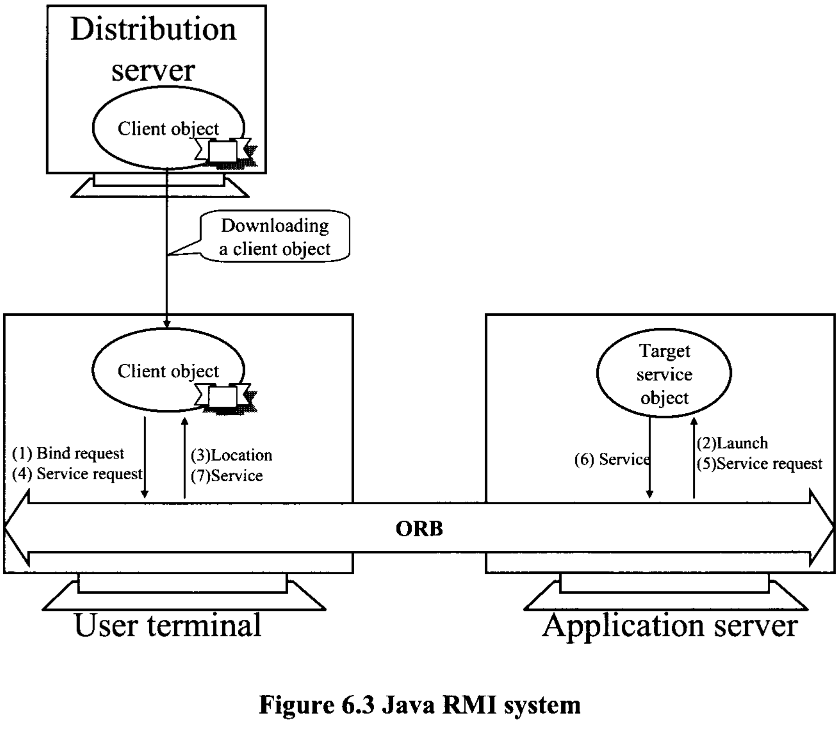


Figure 6.3 Java RMI system

(1) The client object sends a "bind" request to the ORB on the user terminal to solve the location of the target service object. More concretely, the client object calls bind() method of org.omg.CORB.ORB class.

(2) When ORB receives the "bind" request, ORBs communicate with each other to seek a location of the target service object. And if the target service object is not executed, ORB on the application server launches the target service object

(3) ORB returns the location of the target service object to the client object.

(4) The client object sends a service request to ORB on the user terminal by using the location.

(5) The ORB on the user terminal transmits the service request to the ORB on the application server. The ORB on the application server sends the service request to the target service object.

(6) The target service object provides service to the client object via ORBs.

(7) The client object receives service via ORBs.

6.3.2 Interceptor

An interceptor is a mechanism for Java RMI system to provide various value-added functionalities.

The interceptor intercepts messages between the client object and the ORB on the user terminal or between the target service object and the ORB and performs certain processing like investigating, inserting, deleting, modifying and so on.

Figure 6.4 shows a typical example of security interceptor to protect messages from the client object to the target service object. In this figure, the interceptor on the user terminal encrypts messages received from the client object and the interceptor on the application server decrypts the encrypted message received via ORBs.

It is note that the interceptor intercepting a bind message is called as "bind interceptor", the interceptor intercepting a service request message between the client object and the ORB is called as "client interceptor" and the interceptor intercepting a service request message between the ORB and the target service object is called as "server interceptor".

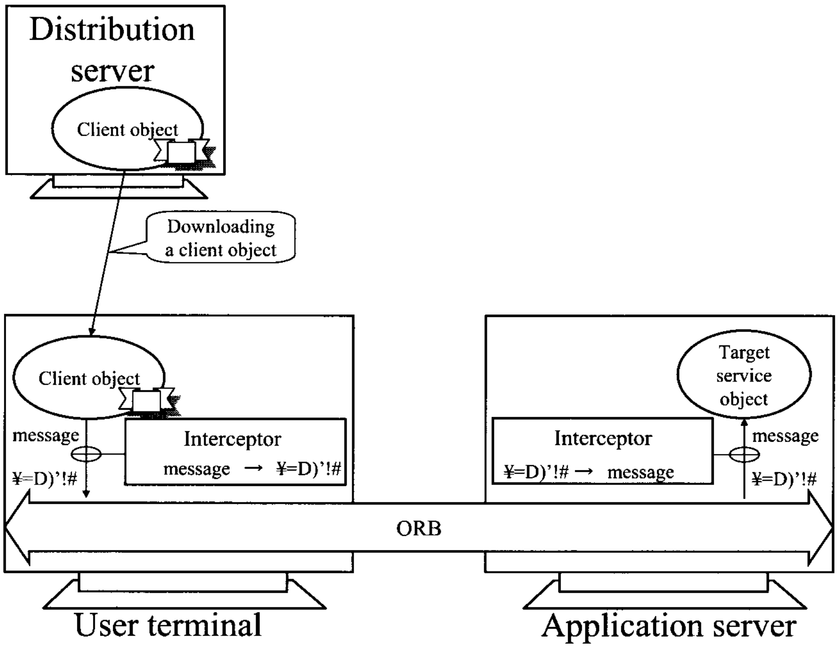


Figure 6.4 Typical example of security interceptor

6.3.3 Java system with SOA

This section describes overview of Java system with SOA (Figure 6.5).

Java system with SOA also contains the distribution server, the application server and the user terminal. In addition, Java system with SOA contains an SOA server.

The SOA server is a server to register and manage a signer's information of the client object.

The application server's administrator is required to make a list of trusted signer, who is allowed to receive the service, for each target service objects as an access control list (ACL) and to store the ACL on the SOA server.

The SOA server makes an access decision by comparing signer's information received from the user terminal and the ACL.

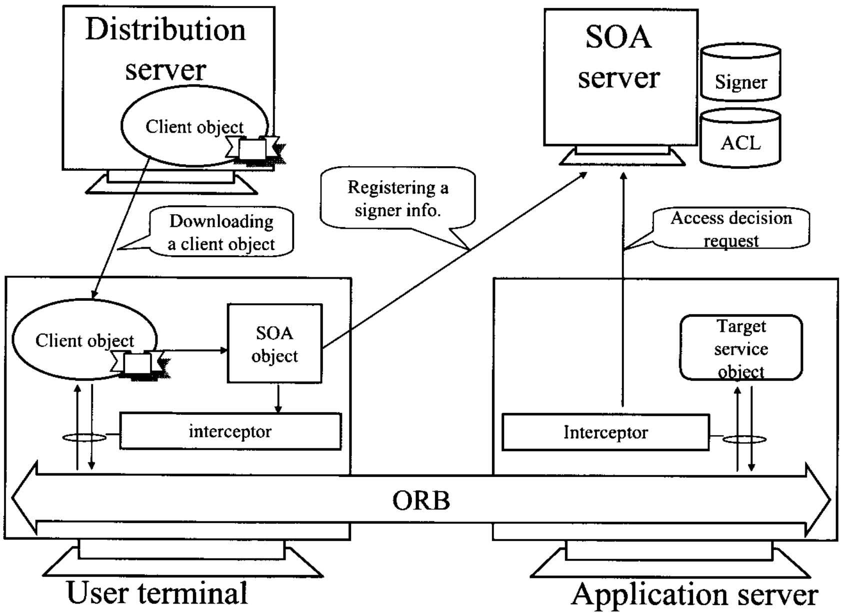


Figure 6.5 Java system with SOA

This thesis proposes centralized access decision by introducing the SOA server because the management of ACLs is easier than distributed access decision model in the multiple application servers' environment even if the SOA server is required to be protected from unauthorized access.

In Figure 6.5, the client object, the ORB, an SOA object, a bind interceptor for SOA and a client interceptor are running on the user terminal.

The SOA object is the object to retrieve a signer information of client object and register it to the SOA server. In addition, the SOA object provides an object identifier (ID) issued by SOA server to the client interceptor.

The client interceptor adds the ID to the service request message when intercepting it.

The target service object, the ORB and a server interceptor for SOA are running on the application server.

The server interceptor for SOA retrieves the ID from the service request message and asks the SOA server to make an access decision when receives the service request message.

It is note that the client object is required to be a signed applet programmed to use SOA mechanism clearly because of registering a correct signer's information during the bind process. If the client object is not signed or not programmed to use SOA mechanism, it does not work in the proposed system.

When the client object is downloaded from the distribution server to the user terminal in the Java system with SOA, the client object performs a bind process at first.

For example, in Figure 6.6,

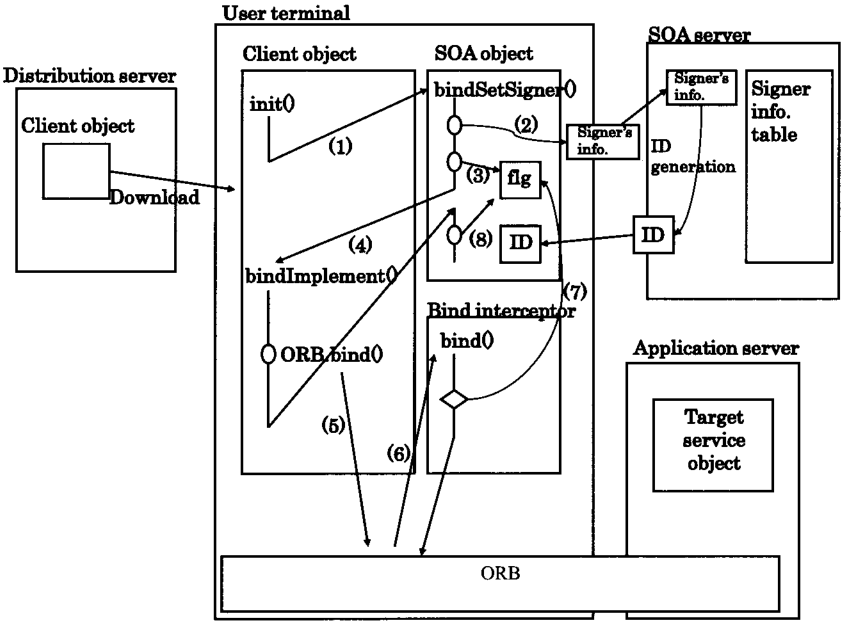


Figure 6.6 Bind process on SOA mechanism

(1) when the bind process is executed in the Java applet method init(), the client object calls a method bindSetSinger() of the SOA object with the object reference of the client object itself (i.e., this) as a parameter.

(2) The SOA object retrieves signer's information from this, and sends it to the SOA server. When the SOA server receives the signer's information, the SOA server generates an ID, returns the ID to the SOA object and stores a pair of the ID and the signer's information into the signer information table.

(3) The SOA object holds the ID received from the SOA server and sets a flag to confirm if it complies with correct SOA procedure,

(4) And the SOA object calls a method bindImplement() of this that is the method for the bind process. This is to prevent from registration of fake signer by calling a method bindSetSinger() with another authorized object as a parameter.

(5) The client object calls a method bind() of ORB in the method bindSetSinger().

(6) The ORB calls a method bind() of the bind interceptor.

(7) The bind interceptor checks if the flag is set.

(8) After finishing the bind process (i.e., the SOA object receives a return value of bindImplement()), the SOA object sets the flag off. This is to prevent other client object from using the flag.

Then the client object calls the target service object as shown in Figure 6.7. Namely,

(1) the client object sends a service request message to the ORB on the user terminal.

(2) The ORB calls a method prepare_request() of the client interceptor with the service request message as a parameter.

(3) The client interceptor retrieves the ID from the SOA object and returns the service request message with the ID. The ORB transmits the service

request message with the ID from the user terminal to the application server.

(4) When the ORB on the application server receives the service request message,

(5) the ORB calls a method receive_reqeust() of the server interceptor with the received message as a parameter. The server interceptor retrieves the ID and a name of the target service object from the message and calls a method check() of SOA server with the ID and the name as parameters to ask the SOA server to make access decision.

(6) The SOA server retrieves a signer's information correspond to the ID from the signer information table and checks if the signer is allowed to receive the service indicated by the given name based on the ACL.

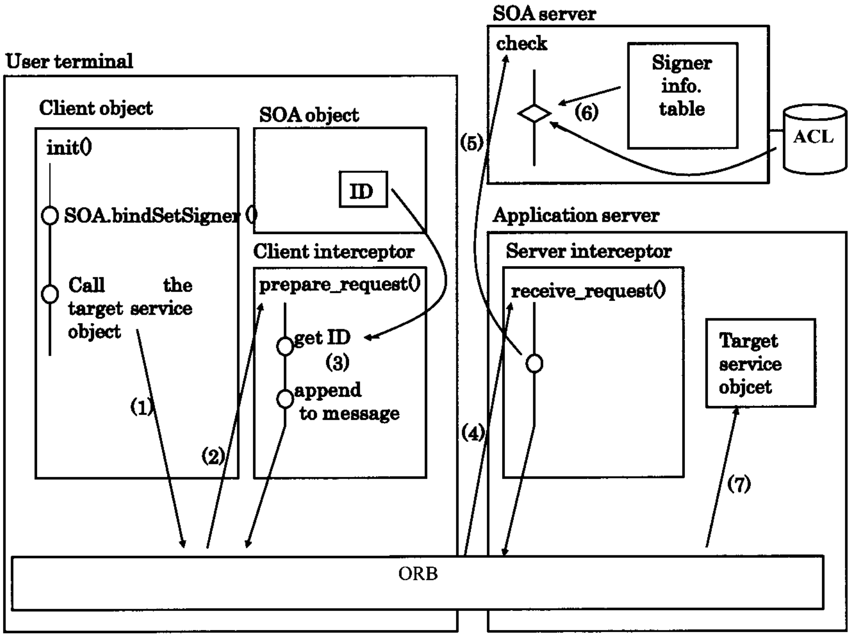


Figure 6.7 Service request process on SOA mechanism

Table 6.1 Comparison between Java RMI system and Java system with SOA

	Type of client object	Permission of execution		Java RMI system	Java system with SOA
		User	Administrator		
(1)	Unsigned applet	-	-	Refuse	Refuse
(2)	Signed Applet	Deny	-	Refuse	Refuse
(3)		Allow	-	**Provide**	**Refuse**
(4)	SOA-aware Applet	Deny	Allow/Deny	Not work	Refuse
(5)		Allow	Allow	Not work	Provide
(6)		Allow	Deny	Not work	**Refuse**

"Allow" means that the user/administrator grants a permission to execute/access.

"Deny" means that the user/administrator does not grant a permission to execute/access

"Provide" means that the system provides a service to the client object.

"Refuse" means that the system refuses to provide a service to the client object

"Not work" means that the client object does not work.

(7) The server interceptor transmits the service request message to the target service object via the ORB if the SOA server returns a positive answer to the server interceptor.

6.4 Evaluation of SOA approach

Table 6.1 shows Comparison between Java RMI system and Java system with SOA.

In the Java RMI system, the "unsigned" applet or the signed applet without user's permission is refused to provide a service ((1) and (2)).

However, if the user permits to execute the signed applet, the system allows to access a target service object and provides a service to the client object ((3)).

It is note that the SOA-aware applet that is programmed to use SOA does not work in the Java RMI system because of lack of SOA object ((4)-(6)).

The Java system with SOA refuses to provide the service to the unsigned applet and the signed applet (i.e., SOA-unaware applets) ((1)-(3)).

In addition, the Java system with SOA refuses to provide the service to the SOA-aware applet if the administrator does not to grant permission even if the user grants permission ((6)).

As the result, the Java system with SOA allows to provide the service to the SOA-aware applet if and only if both the administrator and the user grant permissions ((5)).

Namely, in the Java system with SOA, the administrator is able to control if it provides a service or not. Especially, even if the malicious client object is executed from careless of the user, the Java system with SOA can refuse to provide a service.

6.5 Conclusion

There is a possibility of new problems, which are caused by careless of the developer, the distributer or the user of the mobile code and cannot be prevented by security functions of NGN.

The SOA approach is a solution against one of problems that the malicious signed mobile code makes unauthorized access by acting as a privileged user.

In the SOA approach, the signer information of mobile code is verified not only on the user terminal, but also on the application server. The system implemented the SOA approach provides a service to only mobile codes where the signers are trusted by the administrator beforehand.

Namely, in the Java system with SOA, the administrator is able to control if it provides a service or not. Especially, even if the malicious client object is executed from careless of the user, the Java system with SOA can refuse to provide a service.

Chapter 7
Conclusion

In this thesis, the following security techniques have been studied to address security-related issues on the enterprise network over NGN: (1) the protection of communications over enterprise networks, especially, the protection of communications crossing over NGN and the Internet, (2) a TLS implementation method on cross-certification environment that can reduce a number of certificates to be managed, and (3) an object authentication method that the network platform verifies a digital signature of an object code which is running on a home gateway to protect the enterprise network from malicious coding.

The existing work related with those topics was summarized in Chapter 2. At first, some related work concerned with NGN security was mentioned. Next, some related work concerned with mobile code security was mentioned.

In Chapter 3, a model of the enterprise network over NGN dealt with in this thesis is described. The model is the network that several LANs in headquarters, branch offices and SOHOs are connected with NGN and the Internet. Chapter 3 also described security threats to be solved.

It notes that threats described in Chapter 3 may come from SOHOs if SOHOs are not managed properly. It means that these threats are required to be considered if the enterprise network, where LANs in headquarters, branch offices and SOHOs are directly connected with NGN only, is constructed.

In Chapter 4, a secure session provider service was proposed as a first security technique. The first security technique aims to protect communications crossing over NGN and LANs in cooperation with NGN's call session control function. A secure session provider service consists of a user plane function, which protect communication data, and a control plane function, which control the user plane function in cooperation with CSCF of NGN. And a secure session provider service uses SIP as a protocol of the control plane and TLS or IPsec as a protocol of the user plane. Especially, because a secure session service provider server authenticates all of entities as a trusted third party and it skips a peer entity authentication at the communication session establishment phase, it is able to establish a secure communication session of the user plane quickly.

In Chapter 5, a TLS implementation method for the cross certification model was proposed as a second security technique. The second security technique aims to protect SIP messages by TLS under the interconnection with huge number of PKI environments if enterprises and NGN service providers will operate their own PKI respectively. For the interconnection with multiple PKI environments, there are two models; "multiple trust anchors environment" and "cross certification environment". If there are many CAs, the cross certification environment is useful. However, because TLS is designed for the multiple trust anchors model, TLS will not be able to work efficiently on the cross-certification model. This thesis proposed and evaluated a profile of TLS handshake protocol for the cross certification environment and a TLS implementation method with a delegated certification path discovery and validation server for the cross certification model efficiently.

In Chapter 6, a seamless object authentication (SOA) method was proposed as a third security technique. The third security technique aims to protect the entire enterprise network from malicious mobile codes running on SOHO equipments because it is difficult for a manager of enterprise network to maintain SOHO equipments properly. Although most of existing mobile code execution platforms use digital signature technology to protect from malicious codes, they verify a digital signature of a mobile code only at time when the mobile code begins executing. Once the mobile code executed, the mobile code can access any resources in the enterprise network by the user's privilege. A seamless object authentication proposed this thesis is that a server received an access request from the mobile code verifies a digital signature of the mobile code as a countermeasure of above security problem. If the enterprise network supports SOA mechanism, a network manager is able to control whether to allow access to the mobile code based on the digital signature. Therefore, even if malicious mobile code is executed by careless of the user, the enterprise network is able to be protected.

The security techniques proposed in this thesis are able to achieve secure enterprise network in cooperation with NGN security function.

Figure 7.1 shows an example of the enterprise network over NGN with security techniques proposed in this thesis.

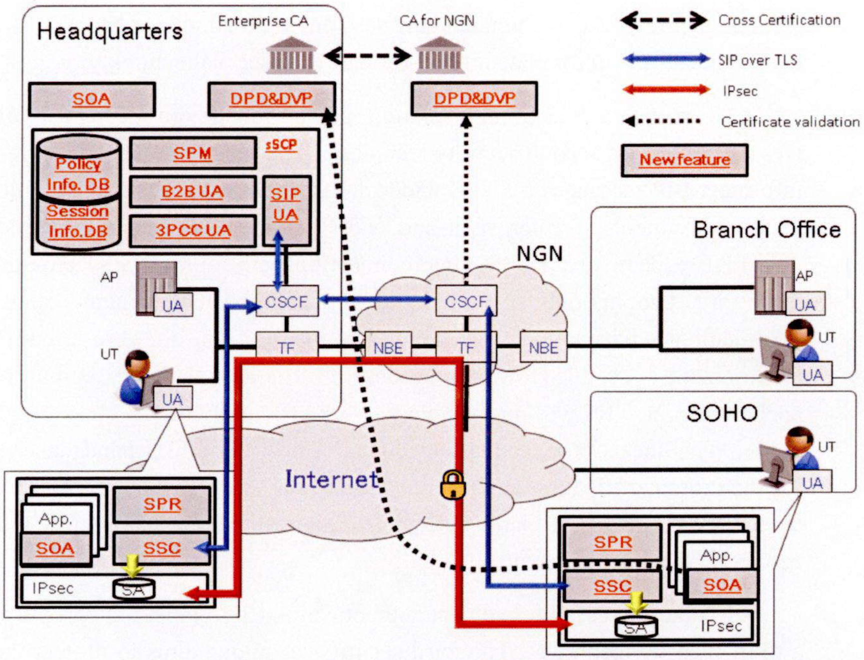


Figure 7.1 Enterprise network over NGN with proposed security techniques

In Figure 7.1, the headquarters has an sSCP server system, an enterprise CA, a DPD&DPV server and a SOA server as new features and all terminal in the enterprise network has an sSCP client function as UA.

And the enterprise CA and CA for NGN have cross-certification each other.

If an application, which can be developed as a signed mobile code, on a UT in SOHO is downloaded and executed, a mobile code execution platform on the UT sends a digital signature of the sSCP client function to the SOA server to retrieve permission.

If the SOA server gives permission, the application can continue to execute. And if it tries to communicate with other application on a UT in headquarters, SSCs on these UTs receives a common SA from the sSCP server in the headquarters via SIP over TLS sessions between UT and CSCF, between CSCFs and between CSCF and sSCP server. And then, these UTs exchange application data via NGN and the Internet under the protection of IPsec.

When establishing a SIP over TLS session between CSCF in NGN and SSC on the UT in SOHO, CSCF verify SSC's certificate by using the DPD&DPV server on the CA for NGN and SSC verify CSCF's certificate by using the DPD&DPV server on the enterprise CA.

For interoperability between enterprise network system and NGN, a standardization of proposed security techniques as a part of user network interface (UNI) of NGN will be required as one of future works.

Appendix A
SIP DIAL-UP method

Kawashima etc. propose that SIP DIAL-UP method which is controlling Internet Key Exchange Protocol (IKE) and IPsec connection with SIP signaling for establishing remote access. ([5])

The SIP DIAL-UP method is applications exchange encryption key by IKE via a UDP session which NGN establishes and assures QoS to encrypt data over enterprise network or home network by IKE and IPsec.

The SIP DIAL-UP method is for remote access to an enterprise network or for home device control from the outside. Namely, the SIP DIAL-UP method is for establishing a virtual private network (VPN) connection with certain VPN gateway (Figure A.1).

If the SIP DIAL-UP method is used for establishing communications between numerous numbers of terminals, following problems will be occurred.

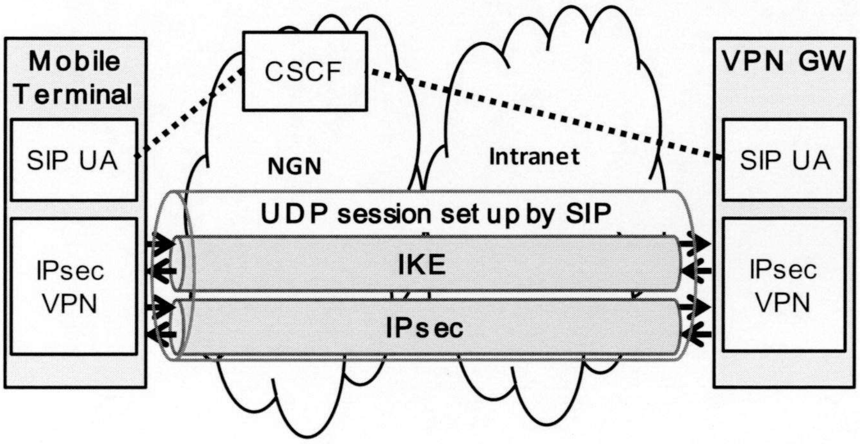


Figure A.1 SIP DIAL-UP method

(1) Management of authentication information

The SIP DIAL-UP method requires managing not only authentication information to access NGN, but also authentication information for each access peers. There is not serious if remote access to an enterprise network or home device control. It requires managing only two authentication information because communication peer is only one in such cases. However, in the cases that the enterprise with multiple branch office constructs an enterprise network or of peer-to-peer (P2P) communication application, the SIP DIAL-UP method requires managing authentication information, which equals to the number of communication peers.

(2) Cost of session establishment

The SIP DIAL-up method establishes a media path by SIP and exchanges an encryption key over the media path to establish a secure session. Key exchange requires the authentication of communication peer and the negotiation of available algorithms for encryption, message authentication and so on. It takes time until application data can be sent over the established secure session.

Acknowledgement

I would like to express my sincerely appreciation to continued support of my supervisor Professor Teruo Higashino of Osaka University through trials and tribulations of this Ph.D thesis. I would like also thank his encouragement and invaluable comments in preparing this thesis.

I am very grateful to Professor Masayuki Murata, Professor Koso Murakami, Professor Makoto Imase and Professor Hirotaka Nakano of Osaka University for their invaluable comments and helpful suggestions concerning this thesis.

I am very grateful to Mr. Takahiro Fujishiro, Dr. Susaki Seiich, Dr. Akihiko Takase, Mr. Shinnichi Irube and Dr. Kazuyoshi Hoshino of Hitachi who have provided many valuable comments and discussions.

I am also very grateful to Dr. Masato Terada and Dr. Kazuo Takaragi, Professor Satoru Tezuka of Tokyo University of Technology, Professor Ryoichi Sasaki of Tokyo Denki University who for their support.

Thanks go to all the members of Higashino Laboratory of Osaka University and of Systems Development Laboratory of Hitachi for their helpful advice.

A part of this thesis was supported by the Ministry of Internal Affairs and Communications, Japan as the “R&D of technologies for advanced network authentication - technologies for secure communication platform based on user authentication”.

A part of this thesis was supported by the National Institute of Information and Communication Technology (NICT) of Japan as the “Research and Development of Next Generation Network Basic Technology.”

Finally, I would like to thank my family, especially my wife for her help and understanding.

Bibliography

[1] A. Tsutsui: "Management Architecture and Distribution Framework for Home Network Services," ITU-T NGN Technical Workshop (2005) https://www.itu.int/ITU-T/worksem/ngntech/presentations/s4-tsutsui.pdf

[2] T. Fukazawa, T. Nisase, M. Kawashima, T. Hariu and Y. Oshima: "Safe and Secure Services Based on NGN", IEICE Trans. on Information and Systems, Vol.E91-D, No.5, pp.1226-1233 (2008)

[3] S. Kent and R. Atkinson: RFC2401, "Security Architecture for the Internet Protocol," IETF (1998)

[4] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler: RFC3261, "SIP: Session Initiation Protocol" , IETF (2002)

[5] M. Kawashima, S. Mizuno, and J. Kato: "Architecture for broadband and mobile VPN over NGN", Innovations in NGN: Future Network and Services 2008 (K-INGN 2008), pp.187-194 (2008)

[6] http://flets.com/pdf/ip-int-flets-1.pdf (in Japanese)

[7] ETSI TS 187 001, " Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN);NGN SECurity (SEC);Requirements", ETSI (2006)

[8] Recommendation ITU-T Y.2701, "Security requirements for NGN release 1", ITU-T (2007)

[9] L. Zhang and K. Zhao: "Study on Security of Next Generation Network", IEEE International Conference on Service Operations and Logistics, and Informatics, 2008, pp. 538-541 (2008-10)

[10] Z. S. Khan, M. Sher, K. Rashid and I. Razzak: "Towards Security and Enrichment of the IP Multimedia Subsystem Based Multiparty Conference", Proceedings of the International Multi Conference of

Engineers and Computer Scientists 2009 (IMECS 2009), Hong Kong (2009)

[11] A. Awais, M. Farooq and M. Y. Javed: “Attack analysis & bio-inspired security framework for IP Multimedia subsystem”, GECCO '08: Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation, pp.2093-2098 (2008)

[12] ITU-T Recommendation I.322, “Generic protocol reference model for telecommunication networks,” ITU-T (1999)

[13] ITU-T Recommendation X.1152, “Secure end-to-end data communication techniques using trusted third party services”, ITU-T (2008)

[14] T. Kaji, O. Takata, K. Hoshino, T. Fujishiro and S. Tezuka: “A Model for Establishing Secure Communication in Secure Service Platform”, IPSJ SIG Notes, CSEC-033, pp.151-156 (2005) (in Japanese)

[15] T. Kaji, K. Hoshino, T. Fujishiro, O. Takata, A. Yato, K. Takeuchi and S. Tezuka: “TLS handshake method based on SIP”, Proc. of the International Multi Conference on Computer Science and Information Technology, pp.467-475 (2006)

[16] ETSI, Draft ETSI ES 202 504-2 v0.0.5: “Open Service Access (OSA); Parlay X Web Services: Part 2: Third Party Call (Parlay X 3),” ETSI (2007)

[17] T. Nisase: “Achieving Security in the NGN”, NTT Technical Review, Vol.20, No.9 (2008) (in Japanese)

[18] E. B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema and D. Gurle: RFC3428, “Session Initiation Protocol (SIP) Extension for Instant Messaging”, IETF (2002)

[19] J. Rosenberg, J. Peterson, H. Schulzrinne and G. Camarillo: RFC3725, “Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)”, IETF (2004)

[20] T. J. Mobray and R. C. Malveau: "CORBA Design Pattern," Jhon Wiley & Son's (1997)

[21] Ida: "Quick Learning Java," KYORITSU SHUPPAN CO., LTD. (1996) (in Japanese)

[22] R. Orfali and D.Harkey: "Client/Java Programming with Java and CORBA," Jhon Wiley & Son's (1997)

[23] S. Oaks: "JAVA Security," O'Reilly (1998)

[24] Sun Microsystems: "JDK1.2 Security Documentation," Sun Microsystems (1998)

[25] Katumura: "Security Measures for Mobile Codes," Nikkei Internet Technology, No.14, Nikkei BP, pp.171-176 (1998) (in Japanese)

[26] H. Kojima and H. Maruyama: "Discussion about Object Signing," Proceedings of the 56th National Convention of IPSJ, pp.364-365 (1998)

[27] Furuyama: "DCOM Guidebook," Ohmsha (1997) (in Japanese)

[28] W. Ernst and J. J. Kottler: "Presenting Active X", Sams.Net Publishing, (1996)

[29] OMG: " CORBAservices: Common Object Services Specification - Security Service Specification," OMG (1998)

[30] W. Ford and M. Baum: "Secure Electonic Commerce," Prentice Hall (1997)

[31] E. Okamoto: "Introduction to the Theory of Encryption," KYORITSU SHUPPAN CO., LTD. (1993) (in Japanese)

[32] S. Garfinkel and G. Spafford: "Practical UNIX Security," O'Reilly & Associates (1992)

[33] Shibamiya, Hama, Fujita, Koubara and Kuba, "Access Control, Security Management," Nikkagiren (1993) (in Japanese)

[34] H. Kojima and H. Maruyama: "J Kojima, Maruyama•Java Package System," Proceedings of the Computer Security Symposium '98 (CSS'98), pp.171-176 (1998) (in Japanese)

[35] L.Gong: "New Security Architectural Directions for Java," Proceedings of IEEE COMPCON, pp.97-102 (1997)

[36] D. B. Parker: "The Trojan Horse Virus and Other Crimoids," Computers Under Attack: Intruders, Worms and Viruses, pp.544-554 (1990)

[37] V. Neou and M. Pank: "Activex Controls to Go : The Instant Toolkit for Web Site Developers," Prentice Hall (1997)

[38] D. M. Martin, S. Rajagopalan and A. D. Rubin: "Blocking Java Applets at the Firewall," Internet Society Symposium on Network and Distributed Systems Security (1997) http://www.cs.nyu.edu/cgi-bin/cgiwrap/~rubin/block.ps

[39] R. Sasaki: "The Internet security guide," Iwanami (1999)(in Japanese)

[40] T. Dierks and C. Allen: RFC2246, "The TLS Protocol Version 1.0," IETF (1999)

[41] S. Blake-Wilson, M. Nystrom, D. Hopwood, J. Mikkelsen and T. Wright: RFC3546, "Transport Layer Security (TLS) Extensions," IETF (2003)

[42] T. Dierks and E. Rescorla: RFC4346,"The Transport Layer Security (TLS) Protocol Version 1.1 ," IETF (2006)

[43] S. Blake-Wilson, M. Nystrom, D. Hopwood, J. Mikkelsen and T. Wright: RFC4366, "Transport Layer Security (TLS) Extensions," IETF (2006)

[44] ITU-T: Recommendation X.509, "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks," ITU-T (2000)

[45] R. Housley: RFC3280, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", IETF (2002)

[46] M. Myers, RFC2560, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP," IETF (1999)

[47] D. Pinkas: RFC3379, "Delegated Path Validation and Delegated Path Discovery Protocol Requirements," IETF (2002)

[48] T. Freeman, R. Housley, A. Malpani, D. Cooper and W. Polk: RFC5055, "Server-based Certificate Validation Protocol (SCVP)," IETF (2007)

[49] P. H. Baker etc., "XML Key Management Specification (XKMS 2.0)," W3C (2005)

[50] Japanese Ministry of Internal Affairs and Communications, "Governmental Public Key Infrastructure specifications for mutual operation," (2001)
http://www.gpki.go.jp/session/CompatibilitySpecifications.pdf

[51] S. Hane etc., "Speeding up X.509 Certificate Path Validation," 4th International Workshop for Applied PKI IWAP2005 (2005)

[52] A. Alshamsi etc., "A Technical Comparison of IPSec and SSL," Proceedings of the 19th International Conference on Advanced Information Networking and Applications AINA'05, IEEE (2005)

[53] H. Shacham etc., "Client-Side Caching for TLS," ACM Transactions on Information and System Security, Vol. 7, No. 4, pp.553-575 (2004)

[54] OMA, "Wireless Transport Layer Security," OMA (2001)

[55] S. Gadanayak, "TLS specification changes," Personal Wireless Communications ICPWC 2005, pp.399-402 (2005)

[56] G. Apostolopoulos etc., "Securing electronic commerce: reducing the SSL overhead," IEEE Network, Volume 14, Issue 4, pp.8-16 (2000)

[57] P. Persiano, "User privacy issues regarding certificates and the TLS protocol: the design and implementation of the SPSL protocol," Proceedings of the 7th ACM conference on Computer and communications security (2000)

[58] C. He, "Formal analysis of crypto protocols: A modular correctness proof of IEEE 802.11i and TLS," Proceedings of the 12th ACM conference on Computer and communications security CCS '05 (2005)

[59] C. S. Lee and D. Knight: "Realization of the next-generation network," Communications Magazine, IEEE , vol.43, no.10, pp. 34- 41 (2005)

[60] K. Knightson, N. Morita and T. Towle: "NGN architecture: generic principles, functional architecture, and implementation," Communications Magazine, IEEE , vol.43, no.10, pp. 49- 56 (2005)

[61] M. Carugi, B. Hirschman and A. Narita: "Introduction to the ITU-T NGN focus group release 1: target environment, services, and capabilities," Communications Magazine, IEEE , vol.43, no.10, pp. 42-48 (2005)

[62] ITU-T: Recommendation ITU-T Y.2001, "General overview of NGN," (2004)

[63] ITU-T: Recommendation ITU-T Y.2006, "Description of capability set 1 of NGN release 1" (2008)

[64] ITU-T: Recommendation ITU-T Y.2012, "Functional requirements and architecture of the NGN release 1," (2006)

[65] ITU-T: Recommendation ITU-T Y.2014, "Network attachment control functions in next generation networks," (2008)

[66] ITU-T: Recommendation ITU-T Y.2091, "Terms and definitions for Next Generation Networks" (2007)

[67] ITU-T: Supplement 1 to Y-series Recommendations, "ITU-T Y.2000 series – Supplement on NGN release 1 scope," (2006)

[68] ITU-T: Supplement 7 to Y-series Recommendations, "ITU-T Y.2000-series – Supplement on NGN release 2 scope," (2008)

[69] T. Kovacikova and P. Segec: "NGN Standards Activities in ETSI," Sixth International Conference on Networking, pp.76-76, 22-28 (2007)

[70] ETSI: ETSI ES 282 001, "NGN Functional Architecture," Version 3.3.0 (2009)

[71] ETSI: ETSI ES 282 004, "NGN Functional Architecture; Network Attachment Sub System (NASS)," Version 3.3.0 (2008)

[72] ETSI: ETSI ES 282 003, "Resources and Admission Control Sub-system (RACS); Functional Architecture," Version 3.3.0 (2009)

[73] ETSI: ETSI ES 282 007, "TISPAN: IP Multimedia Subsystem (IMS): Functional Architecture," Version 2.1.1 (2008)

[74] B. Gamm, B. Howard and O. Paridaens: "Security features required in an NGN," Alcatel Telecommunications Review , pp. 129-133 (2001)

[75] ECMA Int'l.: "Enterprise Communication in Next Generation Corporate Networks (NGCN) Involving Public Next Generation Networks (NGN)," ECMA International, Technical Report ECMA TR/91 (2005)

[76] ECMA Int'l.: "Next Generation Corporate Networks (NGCN) – General", ECMA International, Technical Report ECMA TR/95 (2008)

[77] Long Zhang and Kai Zhao: "Study on security of next generation network," IEEE International Conference on Service Operations and Logistics, and Informatics2008, pp.538-541 (2008)

[78] R. Marx: "A Service-Oriented Approach on Securing User Plane Traffic between NGN Security Domains," IEEE Wireless Communications and Networking Conference (WCNC) , pp.1-6 (2010)

[79] L. Fang, N. Bitar, R. Zhang and M. Taylor: "The Evolution of Carrier Ethernet Services. Requirements and Deployment Case Studies," IEEE Communications Magazine, Vol. 46, Issue 3, pp.69-76 (2008)

[80] J. Murayama, "Trend of VPN-based communication technology (Invited Talk)," IEICE Tech. Rep., IN2008-48, pp. 27-32 (2008)

[81] E. Rosen and Y. Rekhter: RFC 4364, "BGP/MPLS IP Virtual Private Networks," IETF (2006).

[82] http://www.ntt.com/vpn/ip-vpn/index.html

[83] http://flets.com/vpnwide/

[84] ITU-T: Recommendation ITU-T G.8011.1/Y.1307.1 " Ethernet private line service, " (2004)

[85] http://www.ntt.com/vpn/e-vlan/index.html

[86] R. Cohen: "On the Establishment of an Access VPN in Broadband Access Networks," IEEE Communications Magazine, Volume 41, Issue 2, pp.156-163 (2003)

[87] http://www.ntt.com/vpn/internetvpn/index.html

[88] IDC Japan: JP3073912S "Japan Wide-Area Ethernet and IP-Based VPN Services 2010–2014 Forecast and 2009 Analysis" (2010)

[89] P. Knight and C. Lewis: "Layer 2 and 3 Virtual Private Networks: Taxonomy, Technology, and Standardization Efforts," IEEE Communications Magazine, Vol. 104, Issue 4, pp.124-131 (2004)

[90] M. Finlayson, J. Harrison and R. Sugarman: "VPN TECHNOLOGIES - A COMPARISON," Data Connection Limited (2004)

[91] K. Okada and H. Fuji: "VPN-exchange: A Network Service Providing User-based VPN," Computer Security Symposium 2001 (CSS2001), pp.67-72 (2001) (in Japanese)

[92] K. Ishibashi, M. Ishizuka and M. Aida: "Capacity Dimensioning of VPN Access Links for Elastic Traffic," IEICE Tech. Rep., IN2002-64, pp. 67-72 (2002) (in Japanese)