Efficient VPN construction method using UPnP

Hiroshige Nakahara[*], Yoshitaka Nakamura[**] , and Osamu Takahashi[**]

[*]Graduate school of Systems Information Science, Future University Hakodate, Japan
[**]School of Systems Information Science, Future University Hakodate, Japan
{g2111025, y-nakamur, osamu}@fun.ac.jp

Abstract - There has been a rapid proliferation of small mobile devices such as Android phones and iPhones which can use the Internet. These devices include VPN (Virtual Private Networks) functions as a standard feature. These days, we can economically construct VPNs by using VPN technology of IPsec (Security Architecture for Internet Protocol) and so on. However, setting up a VPN still has issues such as the need for the person doing the set up to be knowledgeable about networks, the setup itself being complex, and the authentication taking a long time even if the VPN is to be used for only a short time. To alleviate these problems, we propose a method for exchanging VPN set-up information between VPN devices automatically and efficiently by using UPnP (Universal Plug and Play). We implemented our proposal in an Android phone and measured the authentication time.

Keywords: VPN, UPnP, IPsec.

1 INTRODUCTION

In recent years, LAN environments have become popular ways of making devices communicate with each other in the home or office. Many LAN environments are built as private communication networks, and they restrict outside access. Therefore, we must build another communication environment for an “outside” device to communicate to a device in the LAN.

We generally use VPN technologies to access devices belonging to a LAN. Currently, IPsec is the most popular VPN technology. It enables us to set up a VPN at low cost. In addition, smart phones such as the iPhone and Android mobile phones have rapidly become prevalent. These devices have VPN using IPsec as a standard feature. The proliferation of these devices will affect the use of VPNs. However, VPNs have issues, in particular they require specialized knowledge to set up and are complicated. Moreover, if we only want to use a VPN for a short time, for example, for checking intra-office network mail or a website, a VPN would take too long to set up and would cause an increase in the authentication time.

In this study, we propose a method for exchanging VPN set up information between VPN devices automatically and efficiently by using UPnP [1]. We implemented our proposal in an Android phone and here show measured results about the authentication time.

2 RELATED TECHNOLOGY

In this study, we used UPnP technology. On-demand VPN is a related technology that is used to build a VPN automatically. The following overviews these technologies.

2.1 UPnP

UPnP is a protocol to control personal computers and peripherals connected to a home network. It is used in broadband routers, TVs, and so on.

The protocol is divided into six main functions: Addressing, Discovery, Description, Control, Eventing, Presentation, and it detects devices, device information, control devices, detect devices’ information by using these functions.

When a home network detects a UPnP compatible device it sends an XML file describing the features and information that can be provided to it. This XML file can be modified to some extent depending on the design and functions of the device.

To control the devices and detect their state, the network has a UPnP device called a UPnP control point device that can operate using SOAP messages [2] written in XML.

2.2 On-demand VPN

The On-demand VPN system builds an IPsec-VPN [3]. The IPsec-VPN authenticates and exchange keys with a partner and builds and manages an SA (Security Association) in order to use IPsec [4][5]. In the past, we needed setup information in order to use the IKE (Internet Key Exchange) [6] on the devices that set up the VPN connections. The on-demand VPN system registers device information with a device management server as a preliminary for this purpose. The VPN management server generates and delivers the necessary information to devices registered in the device management server in accordance with the user’s request to build the IPsec-VPN. This means the devices do not have to have any VPN configuration information, and the VPN management server delivers pre-set information to the devices on demand.

2.3 Issues

The On-demand VPN system has following two issues.

i. It needs a number of servers provided by a third party

The On-demand VPN system needs a third-party device management server for managing devices and a VPN management server for generating and delivering information. Therefore, this system is high cost and constitutes a hurdle to innovation for typical consumers.

ii. Authentication time for constructing a VPN

The On-demand VPN system needs extra sequences before the normal VPN authentication phase because of the characteristics of the system. For example, it needs a device authentication, and it generates and delivers the connection information. Hence, it requires a long authentication time to build a VPN.

3 PROPOSED METHOD

We propose a VPN construction method that avoids the issues described in section 2.3.

The method assumes that mutual authentication between the VPN server and the VPN client takes place when they belong to the same LAN. Second, we assume that the VPN client sends a request for a connection to the VPN server from outside the LAN when the VPN is being built. Third, we assume that the devices have been installed with proper software to use the proposed method and they can use UPnP through this software. Finally, we assume that the VPN is built using a pre-shared key.

Mutual authentication between the VPN server and the VPN client using UPnP is used to generate and exchange VPN connection information between devices. More specifically, the mutual authentication process corresponds to Phases 1 and 2 in IKE.

To build the VPN, the VPN client sends a connection request to the VPN server by using our method's authentication sequences. The following details the mutual authentication sequence.

3.1 Mutual authentication using UPnP

Our method authenticates using UPnP. UPnP has a message by which devices advertise themselves when they want to join a network and it triggers mutual authentication. In addition, UPnP uses plaintext for this purpose. Accordingly, this method encrypts messages using the Diffie-Hellman key exchange algorithm, which is used in IPsec based on common key cipher algorithms such as DES, Triple DES, and AES. Figure 1 shows the details of this process.

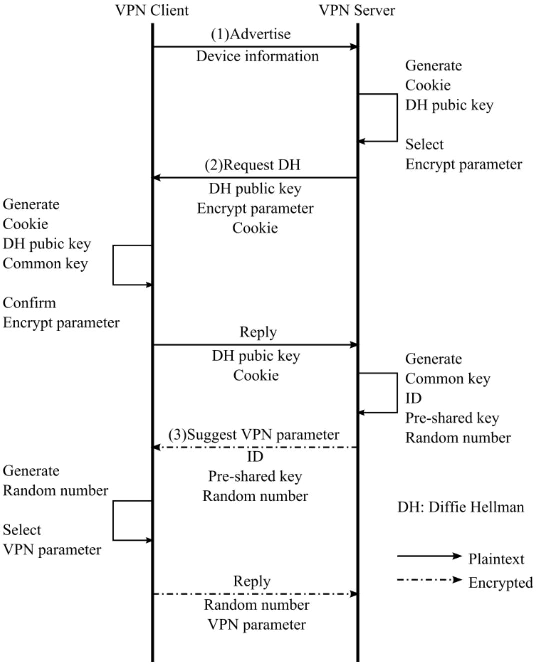

Figure 1: Mutual authentication using UPnP

(1) Advertise

When all UPnP devices join the network, they advertise their presence with an Advertise message. In addition, the advertise message contains where to get a Description file of the device's supported functions. Therefore, the VPN server gets the device's supported functions from the Description file when it receives this message, and it selects those that satisfy a requirement parameter.

(2) Request DH

The VPN server generates the required values in order to send a request to share a key using encrypted communication. First, the VPN server generates the VPN server's cookie by using a random number. Second, it generates a Diffie-Hellman public key for sharing. Finally, it sends a SOAP message containing these values and the parameter selected in (1).

The VPN client generates the required values in order to encrypt communications. First, it generates the VPN client's cookie by using a random number. Second, it generates a Diffie-Hellman public key for sharing. Third, it confirms that the parameters in the received SOAP message from the VPN server are supported. Finally, the VPN client sends a SOAP message containing these values and confirmation information as a reply message to the VPN server.

(3) Suggest VPN parameter

The VPN server generates a common key from the received Diffie-Hellman public key, and the common key is used in the subsequent encrypted communications. Next, it generate an ID in order to uniquely identify the VPN client.

This ID is a hash value generated by combining cookies. The pre-shared key and a number used for constructing the VPN are generated randomly. In addition, the VPN server sends the IP address when the VPN client connects to it from outside the network. These values are encrypted with a pre-selected encryption algorithm using the common key and sent by the VPN server to the client.

The VPN client decodes the received SOAP message by using the common key and selects the most satisfactory requirement parameter in the decoded SOAP message. In addition, the VPN client takes the ID and pre-shared key from the SOAP message and generates a random number in order to build the VPN. It saves these values as VPN connection settings information. Finally, it sends the SOAP message containing the selected parameter and random number as a reply message to the server.

When the VPN server receives the reply message, it saves the associated generated ID together with the pre-shared key.

3.2 Building a VPN

In build the VPN, the VPN client makes a VPN construction request from outside the LAN by using mutual authentication with UPnP. Figure 2 shows the details of this procedure.

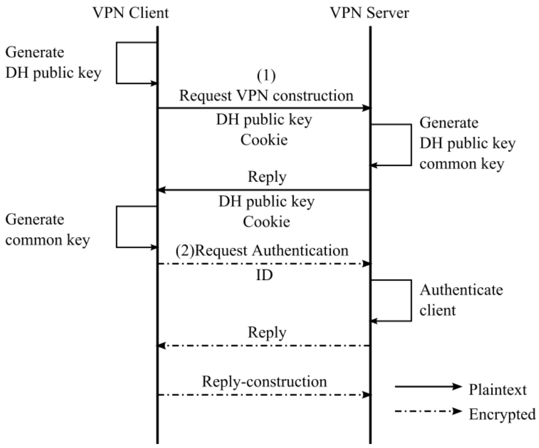

Figure 2: Process of build VPN

(1) Request VPN construction

The VPN client generates a Diffie-Hellman public key for encrypted communication and sends a VPN construction request message containing a Diffie-Hellman public key and Cookies.

The VPN server generates a Diffie-Hellman public key and common key from the received VPN client's Diffie-Hellman public key. Then, the VPN server sends a reply message containing the Diffie-Hellman public key and Cookies to the VPN client. Moreover, it selects the encryption method associated with the Cookies and subsequently uses encrypted communication.

When the VPN client receives the reply message, it generates a common key from the received Diffie-Hellman public key and subsequently uses encrypted communication.

(2) Request Authentication

The VPN client selects the encryption method based on the saved VPN connection settings information through mutual authentication using UPnP. Then, it sends an authentication request message containing its own ID and so on. Additionally, the message is encrypted with the common key, which is MAC from HMAC, by using the saved pre-shared key and random number.

The VPN server authenticates the VPN client by confirming the ID included in the stored authenticate request message. After authenticating the client, the VPN server sends a reply message containing the proper authentication response information.

The VPN client sends a Reply-construction message to the VPN server after receiving the reply message. Thus, VPN built between the VPN client and the VPN server is complete.

4 IMPLEMENTATION AND EVALUATION

We performed a basic experiment on an implementation of the proposed method and a conventional VPN authentication method. Moreover, we did a comparative evaluation of the proposed method, an existing VPN, and on-demand VPN.

4.1 Implementation

We implemented a key exchange method based on ISAKMP [6] using IKE for both the conventional VPN authentication, and the proposed method. We used Java to implement the send and receive ISAKMP messages and used Datagram Socket for the proposed method's messages.

Table 1 shows the encryption parameters used in IKE and the proposed method.

Table 1: Encryption parameters

Encryption Algorithm	AES-CBC 256bit
Hash Algorithm	SHA
Diffie-Hellman group	2048bit MODP
HMAC	HMAC-SHA-1

4.2 Comparative evaluation

Table 2 shows the comparative evaluation of ISAKMP using IKE (below, ISAKMP-IKE), on-demand VPN, and the proposed method in regard to the issues described in Section 2.3. "Yes" means the method is easy to deploy or takes a short time. "No" means there is high cost or a problem with innovation.

Table 2: Comparative evaluation of methods

	ISAKMP-IKE	On-demand VPN	Proposed method
Difficulty of VPN settings	No	Yes	Yes
VPN construction time	Conditional	No	Yes
Server Installation	Yes	No	Yes

First, as to the difficulty of setting up a VPN, the ISAKMP-IKE method essentially requires someone who is an expert in networks and VPNs. In contrast, the on-demand VPN system delivers set-up information for the VPN from a third party and the server provides it to the client automatically. The user side does not need to do anything to set up the VPN. Our method exchanges settings information between the server and client automatically by replacing the VPN software with the proposed method-based software, and so, the user side does not need to do anything to set up the VPN. For these reasons, the on-demand VPN system and proposed method are easier than the ISAKMP-IKE method.

Second, as far as the VPN construction time goes, the ISAKMP-IKE method, which is commonly used in constructing VPNs, will be the criterion for the construction time. Thus, compared with the ISAKMP-IKE method, the on-demand VPN system requires an additional sequence including device authentication, connection information generation and connection information delivery; hence, it takes much longer to set up a VPN. Therefore, on-demand VPN take longest construction time. On the other hand, our method exchanges the required information before the VPN is set up using UPnP, so there is no increase in information when the VPN is constructed. Regarding the sequence size, the ISAKMP-IKE method requires 9 steps for constructing a VPN, whereas the proposed method requires only 5. Therefore, the proposed method has the shortest VPN construction time.

Finally, regarding the Server Installation, the ISAKMP-IKE method requires the VPN server to have a VPN function. The on-demand VPN system requires a number of servers, such as a VPN management server, provided by a third party. Thus, this system is costly and difficult to innovate. Our method is enabled by changing the ISAKMP-IKE method's server software. Therefore, it does not require other servers.

4.3 Basic experimentation and evaluation

4.3.1. Experiment environment

This section describes the experimental results for the authentication time needed for a VPN to be set up by using the key exchange method using ISAKMP-IKE and the proposed method. We define the authentication time as the length of time between when the VPN client sends the VPN connection request to the VPN server and the VPN client finishes sending the last message to the VPN server. Table 3 shows the VPN server and the VPN client that we used. Figure 3 shows the network composition.

Table 3: Experimental environment

	VPN server	VPN client
CPU	AMD Athlon 64 X2 3.00GHz	QSD8250 1GHz
OS	Ubuntu Linux 10.04	Android 2.2.1
Java	1.6.0	NA

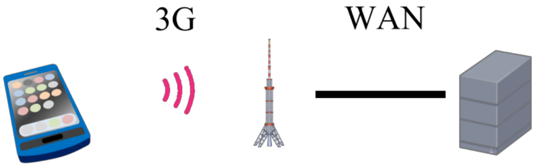

Figure 3: Network composition

4.3.2. Results of experiment

The experiment was based on the content described in Section 4.3.1. We experimented with the ISAKMP-IKE method and proposed method 100 times each. Figure 4 shows the authentication times of each method. Table 4 shows the average time for the 100 experiments.

Table 4: Average authentication time

ISAKMP-IKE	Proposed method
3656.09 [msec]	1884.33 [msec]

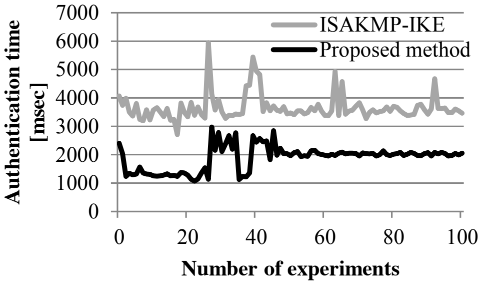

Figure 4: Change in authentication time

From the experimental results, the proposed method can authenticate about 50% faster than the ISAKMP-IKE method. The experiment was conducted in a high-latency network such as a 3G or WAN. Thus, an additional delay is incurred with each communication. Consequently, the difference in the authentication times between the ISAKMP-IKE method and our method would increase and therefore, the usefulness of our method would increase with the delay.

5 CONCLUSION

To reduce the complexity of settings and enable faster authentication for constructing a VPN, we propose a method of exchanging setup information automatically using UPnP and the VPN authenticate sequence. The method reduces the complexity of settings for constructing a VPN through mutual authentication using an UPnP, which changes the required information automatically. In addition, it reduces the authentication sequence by getting the required information before constructing the VPN. An experimental evaluation showed that our method can authenticate faster than existing protocols such as ISAKMP-IKE. Additionally, we found that it is more effective in a high-latency network environment.

We will evaluate the use of our method by implementing the VPN server program. In addition, we will evaluate the security and processing time of our method.

REFERENCES

[1] UPnP Forum, http://www.upnp.org/

[2] Nilo Mitra, and Yves Lafon, SOAP Version 1.2 Part 0: Primer (Second Edition), World Wide Consortium (2007)

[3] Hiroaki Kamoda, Tomoyuki Hoshikawa, Masaki Yamaoka, and Shuichiro Yamamoto, Implementation and evaluation of on-demand VPN system, IPSJ, Vol. 47, No. 8, pp. 2371-2383(2006)

[4] Tatsuya Baba, Mastering IPsec, O'Reilly Japan (2001)

[5] D. Harkins, and D. Carrel The Internet Exchange, RFC2409 (1998)

[6] D. Maughan, M. Schertler, M. Schneider, and J. Turner, Internet Security Association and Key Management Protocol (ISAKMP), RFC2408 (1988)