Key Sharing Method for Flexible Virtual Private Networks

Mirang PARK[†], Akira WATANABE[†], Naonobu OKAZAKI[†] and Tetsuo IDEGUCHI[††]
[†] Information Technology R&D Center, Mitsubishi Electric Corporation
5-1-1 Ofuna, Kamakura, Kanagawa, 247-8501 Japan
[††] Faculty of Information Science and Technology, Aichi Prefectural Univ.
Nagakute, Aichi, 480-1198 Japan
E-mail: mirang@kousoku.isl.melco.co.jp

Abstracts FPN (Flexible virtual Private Network) is a new service which provides secure end-to-end communications. When constructing FPN using secure encryption device, key sharing mechanism is a major technology. In this paper, we propose a key sharing method to apply FPN over the enterprise networks and define communication characteristics for its realization. In this method, all communication entities in a FPN request a session key to key management entity and share the same encryption key. Also, to keep the secrecy of FPNs well, session keys should be updated frequently. An application of this method for secure communication systems will show that inexpensive and flexible VPNs with various communication groups can be constructed easily. It is also shown that the proposed method works effectively in an enterprise network system.

Keywords Virtual Private Networks, Secure Communication Group, Key Distribution, Key Sharing, Data Encryption

1. Introduction

VPNs (Virtual Private Networks) are remarkable new services that provide secure group communications. However, it has a problem that it can not provide a construction method of a secure end-to-end communication group independent from IP subnetworks. Also, users of VPNs communication groups can not join or leave in the group at any time. Therefore, it is necessary to consider a method that could be flexibly construct end-to-end VPNs over the Internet and enterprises networks. In this paper, we consider a constructing method flexible end-to-end VPNs (we call here "FPNs") and a key sharing method between the sender and the receiver for secure end-to-end communications.

Now, Internet standardization is being progressed by the IPSEC (IP Security) WG of the IETF to support these technologies, which defines encryption formats[1][2][3] and sharing mechanisms of encryption keys[4][5].

In [4], it is proposed to distribute a key according to request from each communication entity and to negotiate each other. In this method, all the communication paths have different keys. It may be able to build flexible networks, but settings and management will be complicated because number of keys increases dramatically when a system grows.

We consider a pre-distribution method for FPNs, which distribute a key from management entity to all communication entities beforehand. This is easy to define groups and number of keys stays reasonably small even in a large system. However, there is fear that the entire system will be exposed to danger in the case that someone compromizes the key. To keep the secrecy of the secure communication groups well, session keys should be updated frequently. Also, authentication between communication entities and key management entity is necessary. We propose a key distribution protocol based on user authentication.

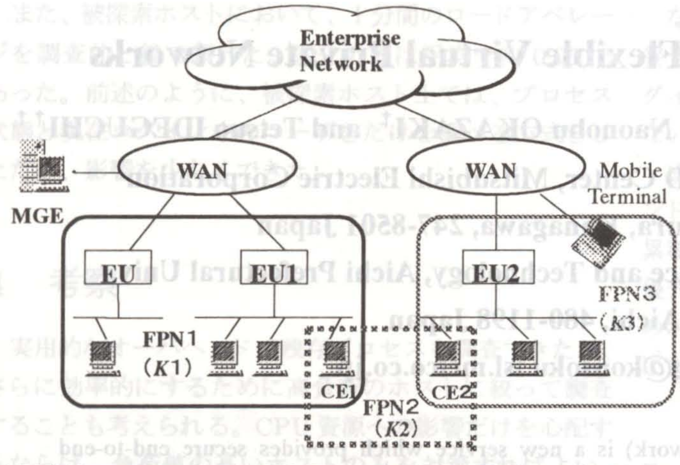
We first consider a construction method of FPNs, and give some problems of the key sharing for FPNs. In section 3, we propose a secure and efficient key sharing protocol. We define key distribution sequence, key searching method and command packet format for realization. Section 4 discusses an implementation and estimation of the proposed method. We realized a simple trial system on the LAN and measured key sharing time.

2. Flexible Virtual Private Networks

VPNs is most commonly implemented in firewalls, allowing organizations to create secure "tunnels" across the Internet. We consider about a construction method that is easy to define a flexible secure communication group that is protected from the various security threats.

2.1 Construction Methods of FPNs

FPN defines a secure communication group by a single encryption key (session key). This provides noteworthy new services that a user can being to multiple secure communication groups. Fig.1 shows a construction methods of FPNs that is applied to enterprise networks. Here, EU (encryption unit) could be implemented such as a hardware-type encryption device or software-type encryption device. In this system, system manager can manages FPNs easily by using a MGE (management equipment) that manages encryption keys and users. It will



MGE: Management Server EU: Encryption Unit Ki: Encryption Key

Fig.1 Construction methods of FPNs.

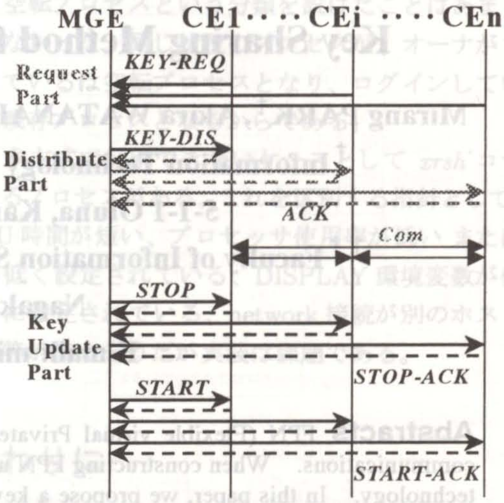


Fig.2 Key distribution sequences.

show that inexpensive and flexible secure communication groups can be constructed easily.

2.2 Problems of Key Sharing

When a constructing these FPNs, a key sharing method between the sender and the receiver for secure end-to-end communications becomes an important problem. We first discuss some problems to be solved on key sharing for FPNs and then briefly state how to solve the problems. Here CEi describes a communication entity, which represents a communication terminal or a set of them, and MGE is a management entity.

(1) *Security problem*

On the key sharing process, there are many types of security threats as below.

- ***Masquerade of MGE***: intruders create false session keys.
- ***Masquerade of CE***: one user pretends to be another user.
- ***Eavesdropping:*** third person monitors the key distribution sequence and analyses the session key.
- ***Data Manipulation***: because of inadequate access control, packet data are modified on the way of transmission.
- ***Replay***: in order to disturb a key distribution action, previous sequences are sent to MGE.

To protect masquerade threats, authentication of CE and MGE is necessary. Also certification of packet data could be performed for checking of modification. Careful definition of packet formats and selection of strong encryption algorithms can solve data manipulation and replay.

(2) *Performance time of Key update*

In order to update session keys, entire systems may be temporarily stopped to distribute new keys to all communication entities. Therefore, shortening the time of the key distribution is very important to keep high performance.

(3) *Conflict of Keys*

In key update process, there is a time when old and new keys exist simultaneously. It is feared that an incorrect key could be decrypt contents of packet. To eliminate conflict of keys, key sharing protocol must have a key elimination mechanism.

(4) *Search of Keys*

If a CE is included in multiple FPNs, CE has a multiple session key. It is necessary to decide a key automatically for each data. To decide a key, key sharing protocol must have a key searching sequence.

3. Key Sharing Method for FPNs

In this section, we propose a key sharing method for FPNs, which solves the above problems. We describe a key distribution sequence, key searching method and packet format as below.

3.1 Key Distribution Sequence for FPNs

We propose a key distribution method based on key elimination mechanism to prevent conflict of keys. We consider that entire systems may be temporarily stopped for distribution of new key to all communication entities. As shown in Fig.2, the sequence consists of a key request part, a key distribution part and a key update part which aims to avoid conflict of keys. Authentication is performed in these parts. We assume that MGE is positioned at physically safe place.

i) CEi→MGE : KEY-REQ
At first, CEi requests a session key to MGE.

ii) MGE→CEi : KEY-DIS
MGE distributes a session key to CEi referring database, which keeps session keys.

iii) CEi→MGE : ACK
CEi sends acknowledge to MGE.

iv) MGE→CEi : STOP, CEi→MGE : STOP-ACK
MGE request the change of the session key to each CEi. Each CEi update a session key and stops data relay function by the old key. Each CEi sends acknowledge to MGE.

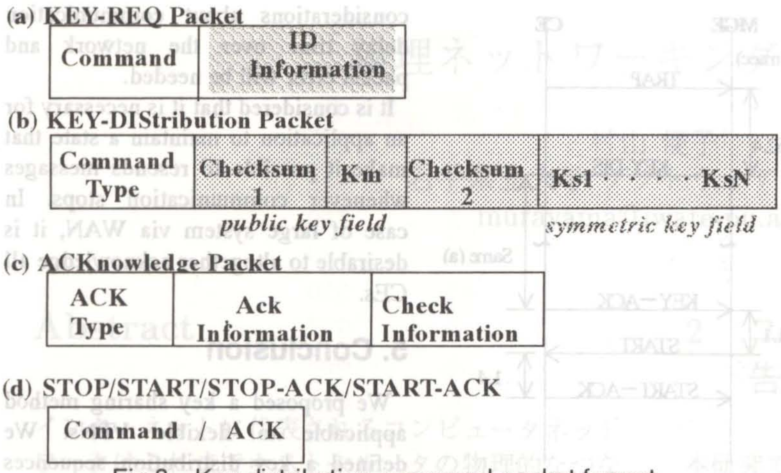


Fig.3 Key distribution command packet format.

v) MGE->CEi : START, CEi->MGE : START-ACK
MGE sends communication restart to each CEi using the new key.

3.2 Key Searching Method

We propose a key searching method that selects a session key automatically for each data. In this method, ECHO／REPLAY packet used of ICMP (Internet Control Message Protocol) to decide a session key. Here, we consider only normal sequence. Other discussions are needed about the case of communication error.

When CE1 requests a secure communication to CE2 in Fig.1, they can decide a session key according to following steps.

step 1:EU1 references a table of session keys.
step 2:EU1 sends key search command packet to CE2.
step 3:EU2 adds own key information to this relay packet.
step 4:CE2 sends acknowledge to EU1.
step 5:EU2 renew own key table according to this packet.
step 6:EU1 register communication path information between CE1 and CE2 to own key table.
step 7:EU1 encrypt a buffered data according to this information, and sends to CE2.

3.3 Command Packet Formats

We define command packet format of the proposed key distribution sequence as shown in Fig.3. In this format, each field defines user data on the UDP. It does not influence any router. In the following, contents of each packet format are briefly described. Only command packets for authentication and key distribution need cryptography.

(1) *KEY-REQ Command*

This is constructed by a header which describes key request and ID information. ID information is encrypted by RSA, which is an asymmetric encryption algorithm. This includes information used for identifying a CE.

(2) *KEY-DIStribution packet*

A hybrid encryption algorithm for authentication is applied for key distribution packet. We combine two types of algorithms, RSA, and MISTY[10], which is a symmetric encryption algorithm. This is efficient for saving time of key distribution.

Encryption field of KEY-DIS consists of public encryption field for public key cryptography and symmetric key field for symmetric key cryptography. Master key (*Km*) in public encryption field is an encryption key for symmetric key field. This is hash value occurred on each KEY-DIS command from MGE. Ks1 ～ KsN means distributed session keys from the stored MGE data base. If a CE is included in two or more FPN, corresponding session keys will be sent simultaneously by using proposed key sharing method. Each field includes a checksum that was calculated before encryption.

(3) *ACKnowledge packet*

This includes acknowledge information and check information. Acknowledge information shows whether the KEY-DIS command has been received by CEs correctly or not. Check information is the hash value of decrypted KEY-DIS packet and used for the authentication of CEs.

(4) *STOP/START packet*

A STOP command stops the relay function of CEs and a START command restarts relay function of CEs.

(5) *STOP-ACK/START-ACK packet*

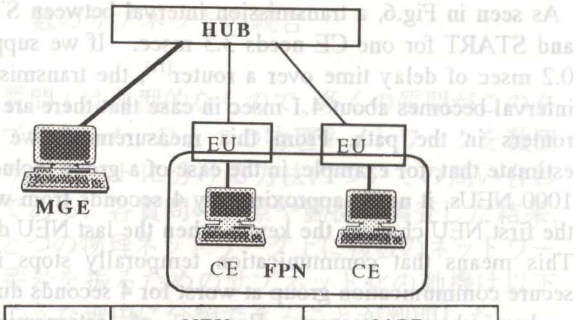
It is a response packet of STOP/START commands.

4. Implementation and Estimation

In this section, we describe a trial system and estimation of the proposed method. We also consider further study based on the result of this estimation result.

4.1 Implementation System

A trial implementation system construction of the proposed method and its specification are shown in Fig.4. In this system, CE is realized by a network encryption unit (EU) and MGE can be realized on workstation (WS).



	NEU	MGE
Interface	10 BASE -T × 2	—
CPU	33MHzRISC	60MHzRISC
OS	Original OS	UNIX
RSA	Original Software	Software Package

NEU:Network Encryption Unit
MGE:Management Entity CE:Communication Entity

Fig.4 A trial implementation system and its Specification.

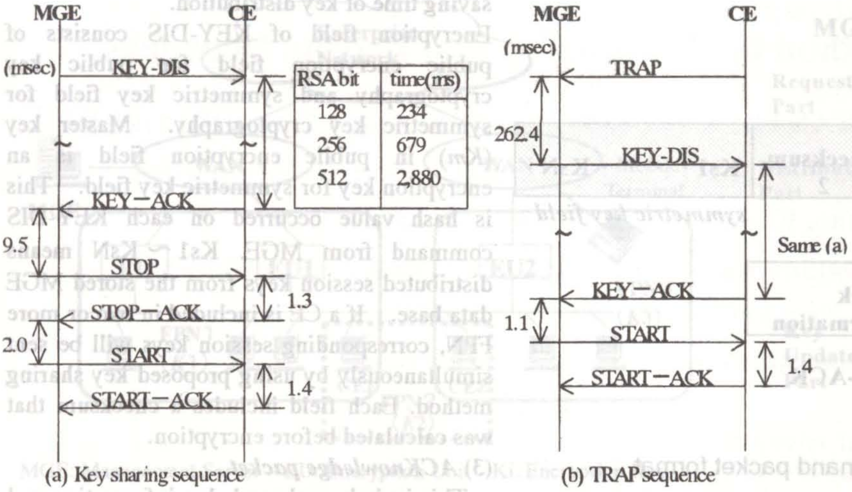


Fig.5 Measured values of key sharing sequences.

NEUs are set between hub and communication terminal. It defines a FPN and encrypts/decrypts a communication packet by using a shared session key. RSA computation by MGE and NEU are realized by software.

4.2 Estimation of Key Sharing Method

We measured time for a key sharing sequence and a KEY-REQ sequence between MGE and one CE. Fig.5 indicates the result. We also measured time when RSA key length is changed as shown in Fig.6. It is shown that time for receiving a key acknowledges message increases exponentially with key length. It is said that the performance of key sharing depends largely on RSA communication time.

4.3 Consideration about Estimation Result

In 3.1, we defined STOP/STRAT commands to solve a problem of key conflict. However, during this sequence, communication between CEs is interrupted. From estimation of the trial implementation system, we can infer the delay time as follows.

As seen in Fig.6, a transmission interval between STOP and START for one CE needs 3.3 msec. If we suppose 0.2 msec of delay time over a router[11], the transmission interval becomes about 4.1 msec in case that there are two routers in the path. From this measurement, we can estimate that, for example, in the case of a group including 1000 NEUs, it needs approximately 4 seconds from when the first NEU changes the key to when the last NEU does. This means that communication temporally stops in a secure communication group at worst for 4 seconds during a key change process. Because of retransmission mechanisms of applications, this does not cause a serious problem in case of LANs or small size enterprise networks.

Because the stop time depends on performance of MGE (Key Management Equipment), it is necessary to select MGE considering system size, kinds of application, key exchange method and investment cost.

In this estimation, we measured the time for normal sequence. In case of a very large system, further considerations about communication delay time over the network and packet error will be needed.

It is considered that it is necessary for an application to maintain a state that make it possible to resends messages whenever communication stops. In case of large system via WAN, it is desirable to altogether acknowledge all CEs.

5. Conclusion

We proposed a key sharing method applicable to flexible VPNs. We defined a key distribution sequences and key sharing protocol. And we applied it to enterprise network. Finally, we constructed a trial implementation system, and measured key sharing time for proposed method. As a result, it is shown that the proposed method works effectively as secure communication systems for LANs or small size enterprise networks.

As future studies, we will consider applying it to large system over the Internet such as an extranet. We will also consider applying it to multicast application system.

References

[1] R. Atkinson, "Security Architecture for the Internet Protocol", RFC1825, Aug. 1995.

[2] R. Atkinson, "IP Authentication Header", RFC1826, Aug. 1995.

[3] R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC1827, Aug. 1995.

[4] P. Karn, et al., "The Photuris Session Key Management Protocol", Internet Draft, draft-simpson-photuris-16.txt, Sep. 1997.

[5] A. Aziz, et al., "Simple Key-Management for Internet Protocol (SKIP)", Internet Draft, draft-ietf-ipsec-skip-07.txt, Aug. 1996.

[6] R. Rivest, "The MD5 Message Digest Algorithm ", Internet RFC1321, 1992.

[7] K. Tanaka, I. Oyaizu, "An Implementation and Evaluation for the Key Distribution Procedure Using ISDN User-to-User Signaling", IEICE Trans.D-1, Vol.J78-D-1, No.6, June 1995 (in Japanese).

[8] Bruce Schneier, "Applied Cryptography", Second edition, John Wiley&Sons, Inc. 1996.

[9] Terry Bernstein et al., "Internet Security for Business", John Wiley&Sons, Inc. 1996.

[10]M. Matsui, "New Structure of Block Ciphers with Provable Security against Differential and Linear Cryptanalysis", the third International Workshop of Fase Software Encryption, Lecture Notes in Computer Science 1039, Springer-Verlag, 1996.

[11]A. Watanabe, et al., "Realization Method of Secure Communication Groups using Encryption and Its Implementation", Trans. IPSJ Vol.38, No.4, p904-914, April 1997 (in Japanese).

[12]M.Park, et al., "Proposal of a Key Sharing Method for Secure Communication Systems", TJCOM98, p113-118, January1998.