A Repeater Encryption Unit for IPv4 and IPv6

Norimitsu Nagashima
Computer Science Department
Columbia University New York, NY 10029
Email: nagashi@cs.columbia.edu

Angelos D. Keromytis
Computer Science Department
Columbia University New York, NY 10029
Email: angelos@cs.columbia.edu

***Abstract*— IPsec is a powerful mechanism for protecting network communications. However, it is often viewed as difficult to use due to the elaborate configuration that is needed to ensure correct (and secure) operation. In this paper, we seek to answer the question of how to build IPsec VPNs without affecting the network assets. We exploit "repeater-encryption", which is similar to the IPsec bump-in-the-wire mode of operation. Our IPsec encryption unit works at Layer-2 of the network stack and does not encrypt control packets that are used for routing, address resolution and resource reservation. Although this is fairly straightforward for IPv4 networks, IPv6 introduces several new features and messages that complicate the operation of such a box. We report our findings of implementing transparent, repeater-based IPsec protection for IPv4 and IPv6. Our approach requires no configuration changes to other devices in the network, making it an attractive mechanism for security network traffic. We discuss the features of our IPsec encryption unit and show how it adapts to IPv4 and IPv6 networks. We also implement our approach on the OpenBSD IPsec stack to demonstrate its feasibility. We show that our transparent IPsec box can easily support speeds in excess of 100Mbps.**

I. Introduction

As a result of the widespread use of the Internet for sensitive as well as everyday tasks, security threats such as data modification and eavesdropping have become a major concern. The IPv6 security architecture itself requires IPv6 nodes to implement the IPsec protocol suite, which provides prevention against some of these threats. However, IPsec may be unwelcome for administrators of enterprise networks because they cannot watch a user's behavior when data leave the user terminal encrypted; such administrators may be required by law (*e.g.*, SEC regulations in the United States) to archive all such traffic for an extended period of time. On the other hand, configuring all nodes of a large enterprise network to use IPsec correctly (and securely) can be a difficult and time-consuming task. One solution for both types of environments is to use a separate IPsec encryption unit under the control of the administrator, to protect traffic of other network elements.

Keromytis, Ioannidis and Smith implemented the IPsec protocol suite for the Linux, OpenBSD and NetBSD [1], focusing on the implementation and its performance on regular hosts and routers. Their system box works on layer-3 of the network stack, as is common for many modern firewall and VPN units. However, IPsec units that work on Layer-3 have some disadvantages when placed into existing networks. One is that routers connect networks which have different network address, so address reconfiguration is sometimes necessary. The other is that routers do not transfer packets that have IPv6 link local address. For instance, consider a number of hosts in the same IPv6 network. They can communicate with other hosts on the same link using IPv6 link-local addresses. If we use IPsec units (that work at layer-3 of the network stack) between hosts to build IPsec VPNs on the same network, we would require IPv6 global address or IPv6 site local address to communicate with other hosts beyond the IPsec units. Address reconfiguration is necessary because IPv6 global address and IPv6 site-local address are not configured automatically.

Keromytis and Wright extend the previous work and propose the IPsec bridge [2]. Their bridge encapsulates Ethernet frames inside IPsec packets. Address reconfiguration is not necessary because the bridge works in Layer-2. However, if routers are in the network between the bridges, routers cannot recognize control packets, because all frames (including control packets) are encapsulated. Routers cannot receive IPv6 control packets such as Neighbor Discovery and Listen Discovery sent by nodes behind the IPsec bridge. This affects routing and the creation of multicast groups in IPv6. The bridge also encrypts Neighbor Advertisement messages. After it decrypts the message, the Hop limit in the IPv6 header is decremented. IPv6 nodes which receive the message drop it because RFC2463 [3] specifies that the Hop limit in the Neighbor Advertise message should be 255, causing address resolution problems in IPv6 networks.

Prevelakis and Keromytis propose portable computing elements [4]. They focus on the system architecture, operation and reliability. Their system works at Layer-2 and can be inserted between the mobile workstation and the network. It works as a firewall and an IPsec gateway to provide individualized security. However, they investigated their system in the context of IPv4 networks. We show that such a device needs to be more careful about which IPv6 packets it protects with IPsec and which must be simply forwarded through.

To use a transparent IPsec device without affecting the functionality of other network nodes in an IPv6 environment, we propose a repeater encryption unit. It works at Layer-2 and does not encrypt IPv6 control packets that are used for exchanging routing information, address resolution and resource reservation in IPv4 and IPv6 networks. This makes it transparent to networks and allows secure communication without changes of host or router configuration or installation of software to existing nodes. To confirm the feasibility of our approach and measure its impact on end-to-end network

communications (which could be significant, given the use of high-grade cryptography), we implemented our repeater-encryption unit and measured its impact in some simple performance experiments.

The rest of this paper is organized as follows: in Section 2 we describe our generic repeater architecture. We study its adaptation to IPv4 and IPv6 networks in Section 3. Section 4 provides some preliminary performance results using our prototype implementation, and Section 5 concludes the paper.

II. Repeater Architecture

We will elaborate on the architecture and show the software module configuration and the packet processing flow.

A. S/W module configuration

Fig.1 shows the S/W module configuration. The Ethernet driver controls two Ethernet ports. We define one port, which connects with the lower network, as the ”plain port” and the other port, which connects to the upper network, as the ”cipher port”. Generally, packets that are received through the plain port are encrypted (and forwarded through the cipher port) and packets which are received through the cipher port are decrypted (and forwarded through the plain port). The Repeater passes the packet to IPsec and the IPv4 or IPv6 protocol suite. TCP/UDP provides the layer-4 functions. Applications such as IKE use the regular socket API. The repeater runs a commodity OS such as OpenBSD or Linux (in our implementation we use the former, due to our familiarity with it).

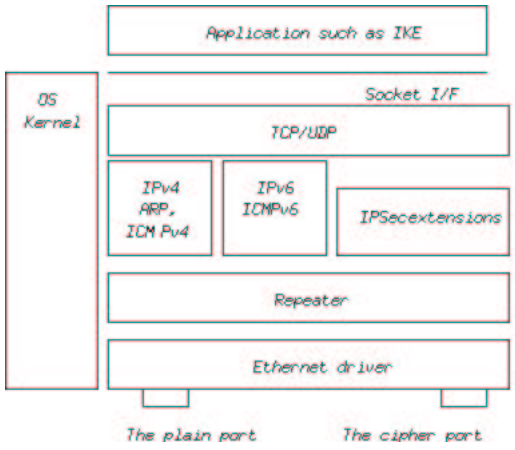

Fig. 1. the S/W module configuration

B. Address configuration

To communicate with other nodes, the encryption unit needs a few addresses. The encryption unit has a MAC address, an IPv4 address and three types of IPv6 address for the two Ethernet ports. The IPv4 address is configured manually or automatically with DHCP. We do not focus on the details because it is a traditional way for nodes supporting the IPv4 protocol suite. In this section, we will mainly describe IPv6 addresses.

A link-local address is configured based on the MAC address. The FP (Format Prefix) of the link local address is defined as the bit-string 1111 1110 10. The link-local address is used for communication with nodes on the same link. In the absence of routers, nodes can communicate with other terminals on the local network using only link-local addresses. The link-local address is also necessary for the Neighbor Discovery Protocol. The link-local address is configured when the encryption unit does not have any other unicast addresses. Routers do not forward packets with link local addresses beyond the local network segment.

The FP of the site-local address is 1111 110 11. The site-local address is equivalent to private addresses for IPv4. Because routers do not transfer the packets which have the site-local address, the site-local address is not reachable from outside networks. The site-local address is not configured automatically. Administrators have to allocate the site-local address manually.

A global unicast address is equivalent to the public IPv4 address. The global routing prefix is provided by the router, ISP or the network administrators assigning the global routing prefix.

C. Overview of the packet processing flow

In this section we describe how our encryption unit works. Fig.2 shows the packet flow when the encryption unit receives the packet through the plain port. Fig.3 also shows the packet flow when the encryption unit receives the packet through the cipher port.

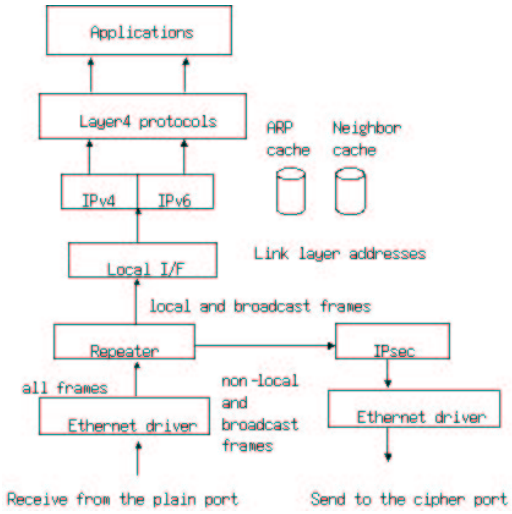

Fig. 2. the packet flow from the plain port to the cipher port

The Repeater is located between the Ethernet driver and the Local I/F and works at the data link layer. The Ethernet driver receives all frames and passes them to the Repeater. The Repeater checks the destination MAC address of the frames. If it is a broadcast or a local frame, which means frames to the encryption unit itself, it is passed to the Local I/F. Then, the frames whose type is 0x0800 are passed to IPv4 and the frames

whose type is 0x86DD are passed to IPv6. Those frames are for the layer-3, the layer-4 or applications in the encryption unit and are not forwarded. Broadcast and multicast frames are copied and are passed from the Repeater to the IPsec module. Non-local frames are also passed to the IPsec module. At the IPsec module the payload of the IP packet is encrypted. This is called ”bump in the wire” (BITW) in the IPsec architecture. In order to get the link layer address of the destination IPsec module looks up the ARP cache for IPv4 or the Neighbor cache for IPv6. Then, the encrypted packet is sent from the cipher Ethernet port.

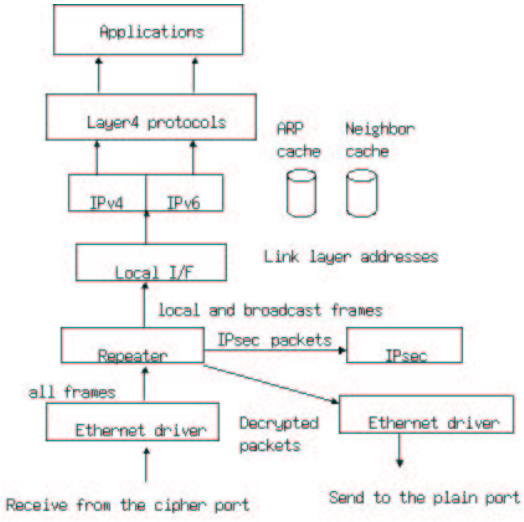

Fig. 3. the packet flow from the cipher port to the plain port

The Ethernet driver on the cipher port also receives all frames and passes them to the Repeater. The Repeater checks the destination MAC address of the frames. If it is a non-local frame, it is passed to the IPsec module. Broadcast and multicast frames are copied and are passed to the Local I/F and sent from the plain port. The local frames except IPsec packets are also passed to the Local I/F. The local frames which contain IPsec packets are passed to the IPsec module to decrypt them. After decryption, packets whose destination IP address is local or broadcast are passed to the Local I/F through the Repeater. Those packets are for encrypted communication for the layer-3, the layer-4 or applications in the encryption unit. The broadcast packets are copied and sent from the plain port. Packets whose destination IP address is non-local are also sent from the plain port. We discuss each module in the next section.

D. The Ethernet driver

The Ethernet driver operates at the physical and data link layer. It controls the LAN controller, for example Intel IXF1104 and FCC in Motorola MPC8255, and makes it receive all frames and send the frames from the Repeater and the IPsec module. If an error occurs during reception or transmission, the Ethernet driver discards the frame. In the IPsec module, data is added to the original packet and put away from the packet. For instance, during encryption the new IP header, the ESP header and trailer are added to the packet; during decryption they are removed from the packet. To avoid the copy of the data, the LAN controller, which is able to transmit packets that consist of several parts of data located in non-contiguous areas in memory, has a performance advantage.

E. The Repeater

The Repeater checks the destination MAC address of the frame and the port at which the encryption units receives and then determines the procedure. The Repeater passes local frames from the plain port to the Local I/F. It passes non-local frames from the plain port to IPsec. If frames are broadcast or multicast, the Repeater copies the frame and passes the source of the frame to the Local I/F and the copied frame to IPsec. On the other hand, the Repeater passes local frames ,which are not IPsec, from the cipher port to the Local I/F. The Repeater passes local frames which are IPsec to IPsec. Non-local frames are passed to IPsec. Broadcast and multicast frames are copied and then the source of the frame is passed to the Local I/F and the copied frame is passed to IPsec.

III. Adaptation to IPv4 and IPv6 networks

We now describe the use of the repeater in IPv4 and IPv6 networks, and in particular how control packets and protocols of various types are handled so as to ensure correct operation of the network.

A. Handling of IPv4 control packets

In order not to affect regular communications, we must be careful in handling of control packets. These are the protocols that the encryption units do not encrypt in IPv4.

[ARP] IPv4 nodes send ARP frames to resolve link layer addresses on the same link. The encryption units must not encrypt the ARP flames because the nodes might not be able to get the link layer addresses. Both non-local frames and broadcast frames are not encrypted.

[ICMPv4] ICMPv4 provides the error and the control information. The ICMPv4 packets have part of the data that caused the error. To protect the part of the data the encryption unit encrypts the ICMPv4 packets, such as Destination unreachable, TTL exceed and Parameter problem etc, if they match the SPD. But the Redirection message is an exception. The Redirect message is send from routers to inform about a better next hop address to reach a destination. In order not to affect routing functions in the network, the Redirect messages should not be encrypted.

[Routing] Routers exchange the routing information with protocols such RIPv4 and OSPFv4. There might be routers between encryption units. If the encryption unit encrypts the routing packets, the encrypted routing information would not be understandable to routers. The routing protocol should not be encrypted at the encryption unit in order not to affect routing functions in the networks.

[RSVPv4] Nodes send the RSVPv4 message to make a reservation for the resources in the network. The encrypted RSVPv4 message would not also be understandable to nodes between encryption units. To make a reservation between end-end communications, the RSVPv4 should not be encrypted at the encryption units.

[IKE] Nodes which support IPsec send IKE packets to establish SAs. The encryption unit does not encrypt the IKE packets, which are sent by other nodes, not to interrupt the establishment of the SA through the encryption units.

B. Handling of IPv6 control packets

We now discuss the handling of the IPv6 control packets. So as not to affect the proper operation of the networks the units are installed in, the handling of control packets is more involved than in IPv4. The following protocols in IPv6 should not be encrypted at the encryption units.

[ICMPv6] ICMPv6 provides the error information (Destination unreachable, Packet too big, Time exceeded, parameter problem), the functions such as Echo, Neighbor Discovery and Listen Discovery. The ICMPv6 packets which contain error information have a part of the data that caused the error. To protect the part of the data the encryption unit encrypts the ICMPv6 packets, such as Destination unreachable, TTL exceed and Parameter problem etc, if they match the SPD. But the Redirection, the Neighbor Discovery and the Multicast listener Discovery message are exceptions. The Redirect message should not be encrypted. This is the same reason as the ICMPv4 redirection is not encrypted. The Neighbor Discovery and the Multicast Listener Discovery message are sent for interaction between nodes. In order not to affect the interaction, they should not be encrypted. We will show the handling of their messages later.

[Routing] Routers also exchange routing information with protocols such as RIPv6 and OSPFv6. RIPv6 and OSPFv6 should not be encrypted, for the same reason that RIPv4 and OSPFv4 are not encrypted.

[RSVPv6] Nodes also send the RSVPv6 messages to make a reservation for the resources in the network. The RSVPv6 messages should not be encrypted. This is the same reason as RSVPv4 is not encrypted.

[IKE] In IPv6 nodes send IKE packets to establish SAs, too. The encryption unit does not encrypt the IKE packets. This is the same reason as IKE on IPv4 is not encrypted.

We show the summary of the protocol handling in TableI.

TABLE I

THE PROTOCOL PROCEDURE

Protocol			Procedure
IPv4	ICMP(1) Redirect Type 5		Not encrypted
	OSPF(89)		
	RSVP(46)		
	UDP(17)	RIP/RIP2(520)	
		RSVP(1698,1699)	
		IKE(500)	
	Others		Depending on SPD
IPv6	ICMP(58)	MLD Type(130,131,132)	Not encrypted
		ND Type(133,134,135,136)	
		Redirect Type 137	
	OSPFv6(89)		
	RSVP(46)		
	UDP(17)	RIPng(521)	
		RSVP(1698,1699)	
		IKE(500)	
	Others		Depending on SPD
ARP			Not encrypted
STP			
Others			Discard

C. Source IP address check

The source IP address of the IPv6 packet is a link-local address, a site-local address or a global address. Routers check the source IP address of the packet and do not forward packets whose source address is a link-local address. Routers also do not forward packets whose source address is a site-local address in IPv6 or a private address in IPv4, to the Internet. To construct the Intranet VPNs in the local network, those packets should be encrypted and transferred by the encryption unit. The encryption units should not check the source IP address. The encryption unit is not a router so it can be possible to encrypt and transfer the packets even though their source address is a link local address or a site local address.

D. IPsec protocol processing

IPsec checks the SPD for the packet which is not a control packet. If it matches the SPD, the encryption unit establishes the SA for the tunnel end point. IPsec stores the packets until an SA is established. If an SA has already been established, the packet is encrypted immediately. IPsec processing is based on RFCs [5],[6],[7].

In the IPsec standard, there are two protocol modes, transport mode and tunnel mode. IPsec hosts have to implement both modes. IPsec gateways, such as routers and firewalls, need the tunnel mode. Our encryption unit is a sort of IPsec gateways so it implements tunnel mode. In tunnel mode, security is applied to the whole IP packet. These are examples of tunnel mode. In tunnel mode, ESP or AH protect the entire inner IP packet including the IP header of the original packet. The encryption unit provides the IPv6 encapsulation for IPv4 packets and the IPv4 encapsulation for IPv6 packets.

E. IP header construction

This section describes how to construct the new IP header and how to handle the fields of the original IP header. The IP source address and destination address in the new IP header are the endpoint of the tunnel. The original IP address in the original IP header is not changed. The TTL is not decremented because the encryption unit is not a router. The IP options or IP extension headers of the original packet are not changed. In the IPv6 header construction, the encryption unit should not decrement the Hop limit.This is the same reason for the TTL in IPv4. The source address in the new IPv6 header depends on the destination address, which means the tunnel end point address. The encryption unit chooses a link local address if the destination address is a link local address. If the destination address is a site local address, it chooses a site local address.

It chooses a global address if the destination address is a global address. When a global address is not configured, a site local address is available if it is configured. When a site local address is not configured, a global address is available if it is configured. When neither a global address nor a site local address is configured, a link local address should be chosen.

F. Source link layer address replacement

After decryption, the encryption unit sends the plain packet to the terminal. A MAC header is added to the packet before sending it. The source address set to the MAC header must be careful. Usually, the encryption unit uses its MAC address for the flames which it sends, but the encryption unit should not use its MAC address for the decrypted packets. This might affect network devices, such as terminals and routers, because the source address of the MAC header is not for the source IP address of the packet. In oder not to affect network devices, the link layer address for the source IP address in the decrypted IPv4 or IPv6 header must be set to the source address of the MAC header. If the network address of the source IP address is not the same as the encryption unit's address, the link layer address of the next hop router is set to the MAC header. By doing this, the encryption unit conceals itself from the terminal which receives the decrypted packet.

G. Multicast packet processing

There are some IETF standards for the Multicast cipher communication [8] [9] [10]. In the RFCs, the key server manages the policy of the cipher communication and distributes the encryption key to the members of the multicast group. The key server controls the system so the availability and the duration of the key server are a big issue. Even through the RFCs are standardized, the multicast cipher communication using the key server has not been widely used yet. So the encryption unit achieves the multicast cipher communication with the static SA. The policies for the multicast SA are set in the SPD and the SA is configured manually. An SA is assigned to ease multicast group. This is an example of the policy:

```
The destination address of the plain packet: FF08::1
The source address of the plain packet: Terminal1
The Encryption Algorithm: MISTY
The Encryption Key: 08081010 02020909 03030707 04040606
The destination address of the new header: FF08::1
The source address of the new header: the encryption unit
```

When the encryption unit receives the packet whose destination IP address is multicast, the IPsec module searches the SPD. If it matches the policy, the packet is encrypted and forwarded. The encryption unit just sends one encrypted packet to the members because the encrypted packets have the multicast address.

H. Path MTU Discovery (PMD)

The packet might be reached through the networks which have the MTU less than 1500 bytes. Fragmentation might occur for full size packets. If fragmentation occurs, it is ineffective for the communication between terminals. The Path MTU Discovery (PMD) is used to avoid fragmentation.

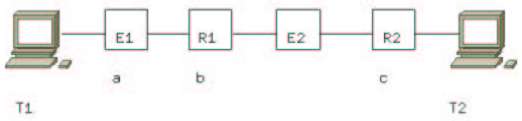

Fig. 4. fragmentation between T1 and T2

This is an example. In the figure E1 and E2 are the encryption units. R1 and R2 are routers and then T1 and T2 are terminals. Fragmentation might occur at E1, R1 (means between E1 and E2), R2 (means between E2 and terminal2). At the E2, the fragmentation does not occur because the MTU of the E2 equals one the E1. It is possible to have the fragmentation between T1 and E1 but it does not matter for E1 and E2. We describe the behavior of the encryption unit for a, b, and c.

a) Fragmentation at the IPsec encapsulation: Fragmentation occurs for long size packets after IPsec encapsulation. The encryption unit sends the Path MTU discovery message to the sender of the packet (T1). The Path MTU discovery message contains the correct MTU. T1 changes the MTU and sends packets which have a suitable packet size. When E1 sends the encrypted packet to E2, E1 stores the information of the plain packet (such as the source address, the protocol, port numbers and the identification in the IP header). That information is used to send PMD to the sender of the packet if fragmentation occurs for the encrypted packets between E1 and E2. After sending the PMD, the encryption unit keeps the value of the MTU for every SA to avoid fragmentation. Refer to b and c.

b) Fragmentation for the encrypted packets: When fragmentation occurs between E1 and E2, the PMD message is sent to the encryption unit (E1). The encryption unit changes the value of the MTU, which is kept in the encryption unit (see a). E1 calculates the correct MTU. The correct MTU is the MTU in the PMD minus the ESP or the AH header size. E1 has to send the PMD message to T1 to inform about the correct MTU. The PMD which is sent to E1 has a part of the encrypted packet that E1 had sent to E2. E1 can not get the sender of the original plain packet because E1 can not decrypt the data which has just a part of the encrypted packet. To specify the sender of the original packet, E1 uses the SPI and refers the stored information mentioned before and then sends the PMD to the sender of the original packet (T1).

c) Fragmentation for the decrypted packets: When fragmentation occurs for the decrypted packet, the PMD is sent to the sender of the packet (T1). E2 encrypts the PMD and sends it to E1. Then, E1 decrypts and transfers it to T1. T1 then sends packets with the correct packet size.

I. Multicast Listener Discovery (MLD)

In IPv6, ICMPv6 is used to discover the multicast listeners. MLD uses ICMP type 130, 131, and 132 (code 0). The MLD has three types of message, the Multicast Listener Query, the Multicast Listener Report and the Multicast Listener Done.

The encryption unit does not encrypt the MLD in order not to affect the construction of the multicast groups.

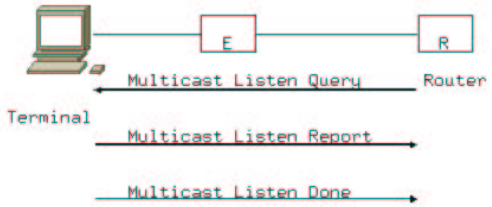

Fig. 5. the MLD procedure

J. Neighbor Discovery (ND)

ND provides the interactions between nodes, such as Router Discovery, Prefix Discovery, Parameter Discovery, Address configuration, Address resolution, Next hop determination, Reachability Discovery, Duplicate Address Detection and Redirect. ND uses ICMP type 133, 134,135,136 and 137. ND has five types of message, Router Solicitation, Router Advertisement, Neighbor Solicitation, Neighbor Advertisement and Redirect. Not to interrupt the interactions, the encryption unit should not encrypt these messages.

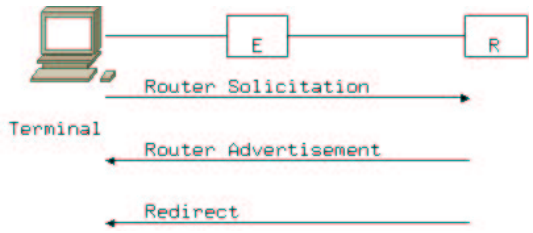

Fig. 6. the ND procedure

IV. Evaluation

To confirm our works, we have implemented the encryption unit on OpenBSD. Fig.7 is a system for confirmation. The system has two repeater encryption units between the server and client. We use IPv6 link local addresses and IPv4 private addresses. All the machines have a Xeon 2.8GHz processor and 1024MB RAM. They are directly connected with 1000BaseT. The encryption units have the different IPsec transforms, ESP DES-SHA1, ESP 3DES-SHA1 and ESP Blowfish-SHA1. First, we make sure that the server and the client can communicate with each other over IPv4 and IPv6 without encryption units. Then, we put the encryption units into the network. They can communicate with each other without any reconfiguration and installation of software. Although they have link local addresses, Their communication through the encryption unit has no problem. This shows that we can apply the cipher communication to anywhere in networks without affecting the network assets.

Next, we measured its performance using netperf[11]. The client is a netperf client and the server is a netperf server. The performance can be seen in TableII.We use software encryptions. They are heavy loads for the CPU but the result shows that it has enough performance to use it on 100Base-TX network.

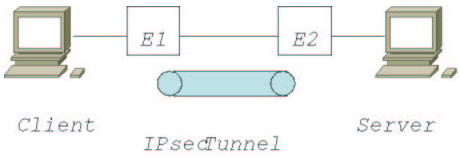

Fig. 7. The system for confirmation

TABLE II
THROUGHPUT THE UDP DATA COMMUNICATION OVER IPSEC

Transform	IPv6(Mbps)	IPv4(Mbps)
Plain communication	956.06	969.21
ESP DES-SHA1	118.36	132.87
ESP 3DES-SHA1	99.85	105.83
ESP Blowfish-SHA1	136.27	159.10

V. Conclusions

We have given an overview of the repeater encryption unit. We also have implemented it on OpenBSD to show the feasibility of our work. As it works in a layer-2 and does not encrypt control packets which are used for routing, address resolution and resource reservation, we can just put the the encryption unit into the network and build IPsec VPNs in anywhere in networks without affecting the network assets. Finally we mentioned the its performance. Although software encryptions are the burden to the CPU, it can be used on 100Base-TX network. The technology of the semiconductor's industry will be rapidly advancing. The higher performance will be achieved if we use a card with a hardware chip.

References

[1] A. D. Keromytis, J. Ioannidis, and J. M. Smith, "Implementing IPsec," in *Proceedings of IEEE Global Internet (GlobeCom)*, August 1997, pp. 1948–1952.
[2] A. D. Keromytis and J. L. Wright, "Transparent Network Security Policy Enforcement," in *Proceedings of USENIX, Freenix Track*, August 1997, pp. 215–226.
[3] A.Conta and S. Deering, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification," IETF, RFC 2463, Dec. 1998.
[4] V. Prevelakis and A. D. Keromytis, "Drop-in Security for Distributed and Portable Computing Elements," *MCB Press Emerard Journal of Internet Research: Electronic Networking, Applications and Policy*, vol. 13, no. 1, pp. 107–115, 2003.
[5] S. Kent and R.Atkinson, "Security Architecture for the Internet Protocol," IETF, RFC 2401, Nov. 1998.
[6] ——, "IP Authentication Header," IETF, RFC 2402, Nov. 1998.
[7] ——, "IP Encapsulating Security Payload (ESP)," IETF, RFC 2406, Nov. 1998.
[8] M. Baugher, B. Weis, T. Hardjono, and H. Harney, "The Group Domain of Interpretation," IETF, RFC 3547, July 2003.
[9] T. Hardjono and B. Weis, "The Multicast Group Security Architecture," IETF, RFC 3740, Mar. 2004.
[10] J. Arkko, E. Carrara, F. Lindholm, M. Naslund, and K. Norrman, "MIKEY: Multimedia Internet KEYing," IETF, RFC 3830, Aug. 2004.
[11] "The Public Netperf Home Page," http://www.netperf.org/.