Highly secure communication service architecture using SDN switch

Masahiro Furukawa† Kouki Kuroda† Takeshi Ogawa† Noriharu Miyaho†

† Graduate School of Information Environment, Tokyo Denki University

1. Introduction
There is an increasing demand for secure communication services that can dynamically reflect user needs. Conventional dedicated services such as an Internet VPN or IP-VPN using IPSec and MPLS have inherent shortcomings, making it difficult for them to dynamically reflect user requirements when faced with limited network resources. In this paper, we propose a state-of-the-art secure communication service architecture. We have previously discussed Disaster Recovery Technology (DRT) [1] which can easily realize highly secure backup services. We propose a robust network architecture achieved by combining DRT and SDN (Software-Defined Network) [2]. It is also quite important for communication services to deal effectively with man-in-the-middle (MITM) attacks, a threat that users are certainly aware of [3]. This paper describes an efficient solution to all these problems.

2. Problems in the conventional method
The IPSec system and IP-VPN systems using MPLS have been developed as conventional secure communication services. However, users will still suffer from eavesdropping in the Internet. For example, in the case of a block cipher using IPSec, various techniques such as a related-key attack or Biclique key recovery for AES [4] are possible avenues for man-in-the-middle (MITM) attacks. In general, secure IP-VPN service is costly compared with ordinary Internet service. In addition, the security level is automatically predetermined regardless of the type of communication. It is difficult to accurately reflect the user-required QoS and security levels.

To improve this situation, we propose an SDN framework using multipath support for each user packet.

3. Proposed secure communication technology
3.1 SDN architecture and its modification
The general three-layer architecture of SDN consists of an SDN switch, SDN controller and SDN application. The SDN switch processes data packets, while the SDN controller manages the SDN switch and selects the data processing method. Centralizing the control layer in the controller makes controlling the data layer simple and flexible. SDN software applications can dynamically control the SDN switch via the SDN controller.

We modified the SDN architecture by adding security-enforcing functions such as stream encryption, spatial random scrambling, fragmentation, duplication and shuffling packets inside the SDN switch by making use of disaster recovery technology [1]. The appropriate number of fragments and routing paths can be flexibly determined according to user requirements by the SDN application.

3.2 Proposed network address translation method
We propose a network address translation method whose objective is not to notify the receiving terminal of the sending terminal’s address-related information at all, in addition to avoiding MITM attacks. This method consists of two stages, as shown in Fig.1. The first stage deals with the handling mechanism at the edge-switches, in which each IP packet corresponding to the connection-oriented terminal is segmented and distributed to multiple routes via the Internet by using DRT.

This makes it nearly impossible for MITM attackers to collect all the IP packets of a message and to decrypt the original information. In the second stage, the predetermined edge-switches connected with the destination can gather together all the related packets belonging to the specific flow, translate the destination host addresses of the egress IP packets to a quasi-random number, and translate the host addresses of ingress IP packets in reverse order to hide the IP addresses of terminals. The address translations are done with a shared key negotiated between domains A and B, which accommodate these terminals. As a result, MITM attackers cannot extract the IP packets of targeted terminals by their inherent IP addresses. Even if the attackers try an indiscriminate attack, they cannot determine which IP addresses are relevant to the target communication.

Fig.1. Proposed IP address translation mechanism

3.3 Proposed method utilizing DRT and SDN
Encryption by DRT is performed in the following steps: spatial random scrambling of file data, subsequent random fragmentation of the file, and the duplication and encryption of each file fragment using a stream cipher code at each encryption stage. Owing to fragmentation and shuffling in the encrypted data processing, an attacker cannot decrypt the message from any of its parts or fragments. As the number of divisions and duplications is adjustable, it is possible to strengthen the security level according to user requests. The physical configuration of the proposed method and the corresponding processing
flow are shown in Fig.2. This mechanism can provide secure communication services by combining DRT and SDN technology. SDN switch \( \alpha \) encrypts the packet transmitted from host terminal A. The fragmented packets are randomly shuffled, and sent via different paths. SDN switch \( \beta \) restores the received packets to the correct order and sends them to host terminal B. Attacker would need to capture and rearrange all raw packets from all the paths in the correct order.

The SDN controller determines the metadata that describe the encryption key, the extent of fragmentation and multiplex reproduction, and the path routes according to the user request.

As shown in Fig.4, end-to-end delay increases in proportion to the number of fragments. When the number of fragmentations increases, the total end-to-end delay increases, but it is not proportional to the number of fragments.

As shown in Fig.5, the switching delay inside switches \( \alpha \) and \( \beta \) is approximately the same. Here delay in switch \( \alpha \) means the elapsed time before finishing the relaying of a packet after receiving it from the source host A side, and delay in switch \( \beta \) means the elapsed time before finishing sending the packet to the destination B side host. This means that the encryption processing time and the decryption processing time are much the same, not dependent on the number of paths.

The most important characteristics of the method are that it can easily increase security strength without terminal host/server side encoding and decoding procedures. We expect this method to be applied to the construction of future secure SDN services.

5. Conclusion

In this paper, we proposed a state-of-the-art SDN service architecture that can reflect user requests easily, dynamically and flexibly. We also proposed a robust network mechanism that can avoid MITM attacks and network eavesdropping by applying a new network address translation method. We implemented an SDN switch to evaluate the performance of the proposed SDN architecture and verified that a secure communication service using the proposed method is realizable.

Acknowledgment

This work has been partially supported by a study (Issue number: 151) of the National Institute of Information and Communications Technology (NICT) of Japan.

Reference


