ComEX

IEICE Communications Express, Vol.12, No.5, 194–199

Design of new CYPHONIC adapter focused on packet sequential processing scheme

Ren Goto[1, a)], **Kazushige Matama**[1, b)], **Chihiro Nishiwaki**[1, c)], **and Katsuhiro Naito**[1, d)]

[1] *Aichi Institute of Technology, Toyota, Aichi 470–0392, Japan*

a) *r0719en@pluslab.org*

b) *matama@pluslab.org*

c) *chihiro@pluslab.org*

d) *naito@pluslab.org*

Abstract: The security policy changed from the traditional "castle and moat" approach to the zero-trust security model due to the recent change in work styles. The authors have been developing the overlay network protocol, called CYber PHysical Overlay Network over Internet Communication (CYPHONIC), to support the zero-trust security model easily. Additionally, we have designed an essential function of a gateway device called a CYPHONIC adapter because the proposed protocol requires devices to install a special client program. The initial implementation suffers from performance degradation of Transmission Control Protocol (TCP) due to order changes of packets during packet handling functions. This paper redesigns the adapter functions focused on the packet processing scheme of the CYPHONIC adapter and confirms the throughput improvement in standard protocols.

Keywords: zero-trust security, overlay network protocol

Classification: Network System

References

[1] S.K. Khaitan and J. McCalley, "Design techniques and applications of cyber-physical systems: a survey," *IEEE Syst. J.*, vol. 9, no. 2, pp. 350–365, June 2015. DOI: 10.1109/JSYST.2014.2322503

[2] Q. Yao, Q. Wang, X. Zhang, and J. Fei, "Dynamic access control and authorization system based on zero-trust architecture," CCRIS '20, Proceedings of the 2020 1st International Conference on Control, Robotics and Intelligent System, pp. 123–127, Oct. 2020. DOI: 10.1145/3437802.3437824

[3] T. Yoshikawa, H. Komura, C. Nishiwaki, R, Goto, K, Matama, and K, Naito, "Evaluation of new CYPHONIC: overlay network protocol based on Go language," 2022 IEEE International Conference on Consumer Electronics (ICCE), pp. 1–6, Jan. 2022. DOI: 10.1109/ICCE53296.2022.9730323

[4] R, Goto, T. Yoshikawa, H. Komura, K, Matama, C. Nishiwaki, and K, Naito, "Design and basic evaluation of virtual IPv4-based CYPHONIC adapter," *Journal of Systemics, Cybernetics and Informatics*, vol. 20, no. 3, pp. 55–63, Oct. 2022. DOI: 10.54808/JSCI.20.03.55

DOI: 10.1587/comex.2023XBL0010
Received January 24, 2023
Accepted February 9, 2023
Publicized February 17, 2023
Copyedited May 1, 2023

[5] D. Siemon, "Queueing in the Linux network stack," *Linux Journal*, vol. 2013, no. 231, Article No. 2, July 2013.

1 Introduction

In recent years, the highly mobile nature of workstyles has caused the migration of essential applications to the cloud. Additionally, the number of Internet of Things (IoT) devices is rapidly increasing and cooperating, and some IoT devices can realize a single service by performing distributed computation [1]. The devices and applications do not exist within a single network protected by a secure "wall" as in the past. Fundamental changes in Internet usage patterns are causing tools such as Virtual Private Networks (VPNs) and hardware boxes to be rethought. They are moving away from the traditional "castle and moat" approach to security to zero-trust security. Zero-trust security is a new security approach due to spreading distributed access to devices. It requires mutual authentication among devices to ensure secure distributed communication [2].

Since zero-trust security requires additional implementation for the security functions, typical systems developers concentrate on their service, not the security model. In some cases, they may handle security as a low-priority function compared to the designated service functions. The authors have been developing the overlay network protocol, called CYber PHysical Overlay Network over Internet Communication (CYPHONIC), as the solution for the easy supporting of the zero-trust security model [3]. CYPHONIC is a communication framework that provides mutual authentication and secure communication between devices. Therefore, developers can concentrate on the implementation of their services.

Our conventional implementation requires installing the device program into the end devices to join our overlay network. On the contrary, some conventional devices (General nodes), such as IoT devices, embedded devices, and dedicated servers, tend to avoid the additional installation of the device program due to the limitation of system and hardware resources or the effect on the system reliability. Therefore, we have proposed a concept of a gateway device called a CYPHONIC adapter to provide CYPHONIC service to general devices. The first concept of the design focused on the virtual IPv4 address assignment and simple packet processing mechanisms where each packet is processed simultaneously [4]. As a result, the prototype performs well for only User Datagram Protocol (UDP) communication and suffers from performance degradation of Transmission Control Protocol (TCP) due to misordering packets.

This paper redesigns the gateway device to propose an ordering packet processing scheme to increase TCP performance. The new design stores an arriving order of incoming packets and processes the packets parallelly. Additionally, since the processing time of each packet is different, it reorders outgoing packets according to the arriving order. For the proof-of-concept with implementation, we evaluated the performance of standard protocols such as UDP, TCP, and Internet Control Message Protocol (ICMP). As a result, it was confirmed that the sequential processing scheme

DOI: 10.1587/comex.2023XBL0010
Received January 24, 2023
Accepted February 9, 2023
Publicized February 17, 2023
Copyedited May 1, 2023

has the same throughput as the conventional implementation without incurring significant overhead. Additionally, it was confirmed that the throughput of TCP was improved, resulting in a significant performance increase.

2 CYPHONIC

2.1 Overview of CYPHONIC

CYPHONIC supports three main functions: communication between IPv4 and IPv6, Network Address Port Translation (NAPT) traversal, and seamless mobility, and it realizes the overlay network for end devices. CYPHONIC comprises a cloud service and CYPHONIC nodes, end devices equipped with the device program of CYPHONIC. CYPHONIC clouds provide three main functions: authentication of CYPHONIC nodes, management of network information where CYPHONIC nodes belong, and instructions for establishing tunnel communication. CYPHONIC nodes enable secure end-to-end communication between devices in cooperation with CYPHONIC clouds.

CYPHONIC node has a constant virtual IP address and a Fully Qualified Domain Name (FQDN) to identify the device. The CYPHONIC node identifies the peer node by its FQDN and processes packets through a device program called the CYPHONIC daemon. Then, it performs seamless communication over our overlay network using virtual IP addresses. This daemon program runs as a background process to prepare the virtual interface for communication from applications and handle packets through the virtual interface.

Since the kernel functions route packets from the application to the virtual interface according to the routing table, the CYPHONIC daemon hooks the packets. The CYPHONIC daemon adds a CYPHONIC header to the virtual IP-based packet and encapsulates it with the CYPHONIC message format. Then, it encrypts the entire message and adds a UDP header because CYPHONIC uses UDP-based tunnel technology. Finally, it adds a real IP address to the physical interface and performs UDP communication over the real network. On the receiver side, it performs the opposite operations for decrypting and decapsulating.

2.2 Overview of CYPHONIC adapter

The CYPHONIC adapter is a gateway device to provide CYPHONIC functions to general nodes without the CYPHONIC device program. It has two network interfaces: the real external interface accesses the Internet, and the real internal interface provides CYPHONIC functions to general nodes.

The CYPHONIC adapter has been implemented as an adapter's daemon (adapter daemon) by leveraging and extending the basic design of the CYPHONIC daemon. The adapter daemon has two main functions for communication over our overlay network: communication and packet processing functions and management functions for connected general nodes. The CYPHONIC adapter assigns a unique virtual IP address to a general node based on the information it manages. The general node uses the assigned virtual IP address on the network interface as a real IP address.

The CYPHONIC adapter implements raw socket and promiscuous mode because it needs to receive and process all packets from general nodes. Therefore, it can

DOI: 10.1587/comex.2023XBL0010
Received January 24, 2023
Accepted February 9, 2023
Publicized February 17, 2023
Copyedited May 1, 2023

encapsulate/decapsulate and encrypt/decrypt processing virtual IP packets received from general nodes. On the contrary, when it receives a decapsulated packet from the adapter daemon, it generates an Ethernet frame and sends it to the general node.

3 Proposed scheme

This paper redesigns a gateway device, the CYPHONIC adapter, as a solution to provide CYPHONIC functions to general nodes without the CYPHONIC device program. The conventional CYPHONIC adapter designed the virtual IPv4 address assignment functions and simple packet processing mechanisms where each packet is processed simultaneously. Therefore, it isn't easy to guarantee the sequence of packets due to parallel processing. As a result, we have obtained good performance results for only UDP. On the contrary, the conventional adapter suffers from performance degradation due to misordering packets. In TCP communications, congestion control algorithms and retransmission handling mechanisms are highlighted, performing it challenging to achieve sufficient throughput. Therefore, it requires to be redesigned based on the packet ordering mechanism.

Figure 1 shows the new transmission process of packets in the CYPHONIC adapter. Since the CYPHONIC uses a capsulation scheme to convey an IP datagram, it limits the size of packets from general nodes up to the maximum transferable size (Maximum Transmission Unit (MTU) size − CYPHONIC header size). Since the CYPHONIC daemon is assumed to process packets individually, the Generic Receive Offload (GRO) [5], which functions as an offload for the Network Interface Cards (NICs), is turned off. As a result, the adapter daemon can process individual fragmented packets.

When the internal interface receives packets from general nodes, it stores them

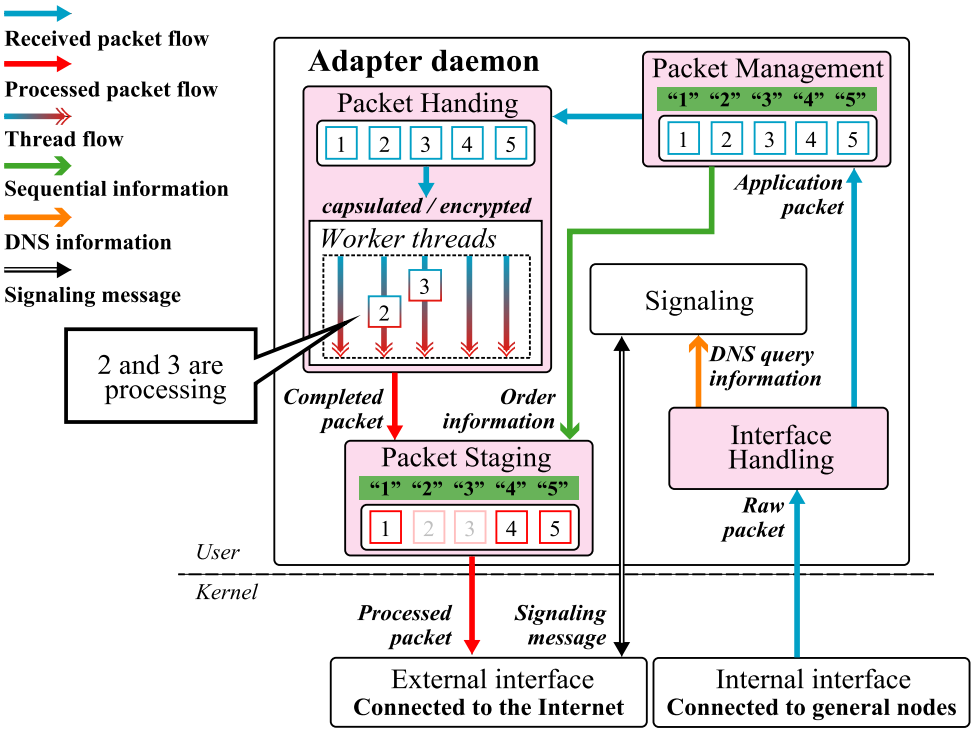

Fig. 1. Sequential processing scheme of outgoing packet

DOI: 10.1587/comex.2023XBL0010
Received January 24, 2023
Accepted February 9, 2023
Publicized February 17, 2023
Copyedited May 1, 2023

in its internal memory. Since it does not perform the packet reassembly process, the CYPHONIC daemon receives the packets individually through a raw socket. It is well known that TCP throughput suffers from out-of-sequence packets. Therefore, the CYPHONIC adapter should store the sequence of received packets when it receives them.

When the interface handling module receives packets, it classifies DNS messages and data packets. It forwards the DNS messages to the signaling module and transfers the data packets to the packet management module. The packet management module stores the sequence order of the received packets. It also transfers the packets to the packet handling module to assign them to some worker threads. Since the processing period of each thread depends on the Operating System (OS) status, the order of process completion may differ from the order of the received packets. Therefore, the packet staging module transmits the processed packets in the order of the received packets by referring to the packet management module. As a result, the order of incoming and outgoing packets can always be maintained.

4 Performance evaluation

Figure 2 shows the verification environment to evaluate the throughput of standard protocols. We prepared a closed network and deployed CYPHONIC cloud services, CYPHONIC adapters, and general nodes. In this verification, we measure the throughput between general and CYPHONIC nodes through the CYPHONIC adapter. They evaluate two scenarios: a case in which a general node communicates with a CYPHONIC node using the proposed CYPHONIC adapter and a case in which a general node communicates with a CYPHONIC node using the conventional CYPHONIC adapter. These two scenarios were conducted using the same verification environment. We used standard protocols such as UDP, TCP, and ICMP as the target of the throughput evaluation. The general node uses macOS Monterey version 12.2 8 GB Dual-core 2.20 GHz Core i7-5650U, and the CYPHONIC adapter and the CYPHONIC node use Raspberry Pi 4 Model B 4 GB Quad-core 1.5 GHz Broadcom BCM2711.

Table I shows the measurement results in two scenarios. We used the ICMP to

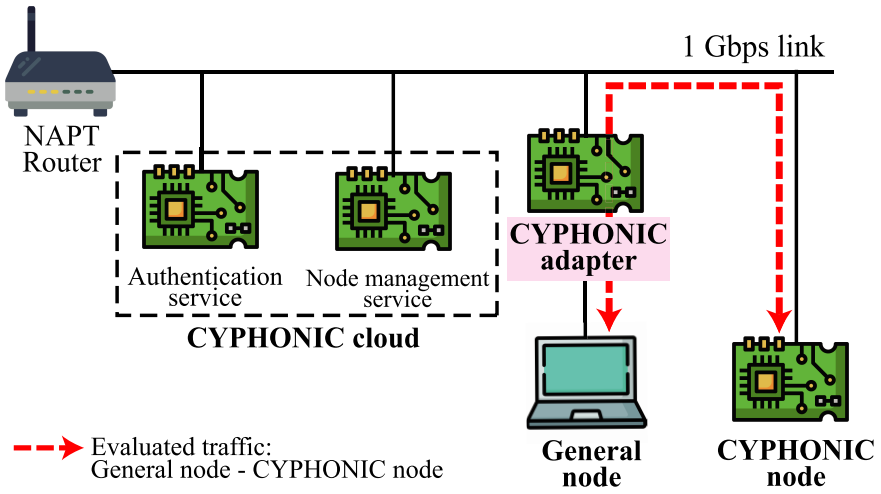

Fig. 2. Evaluation environment

DOI: 10.1587/comex.2023XBL0010
Received January 24, 2023
Accepted February 9, 2023
Publicized February 17, 2023
Copyedited May 1, 2023

Table I. Evaluation results (20-time measurements for each item)

A. Proposal system			
Round-trip time (average)	UDP throughput	Jitter	TCP throughput
3.27 ms	30.0 Mbps	0.50 ms	37.3 Mbps
B. Conventional system			
Round-trip time (average)	UDP throughput	Jitter	TCP throughput
3.47 ms	29.7 Mbps	0.40 ms	1.68 Mbps

confirm the reachability between both nodes and to measure the round-trip time. The evaluation results showed that the overhead of the round-trip time in the CYPHONIC adapter is about 3.27 [ms]. The measurement results confirmed that the proposed method does not cause significant processing delays compared to the conventional CYPHONIC adapter.

For the throughput evaluation, we used iperf3 to measure the performance. We set the 30 Mbps bandwidth for UDP communication to evaluate the processing throughput of the CYPHONIC adapter. The CYPHONIC adapter could process 30 Mbps throughput with low jitter. The performance evaluation results in TCP showed that the proposed method achieved approximately 22.2 times that of the conventional method. Therefore, the proposed processing scheme preserves the ordering rules of individual packets received from the general node and can process fragmented packets correctly. As a result of the performance evaluation, the new CYPHONIC adapter handled about 30 Mbps in both cases of UDP and TCP sufficiently. Typically, High Definition (HD) quality video streaming requires about 5 Mbps. Therefore, the new CYPHONIC adapter implementing the proposed processing scheme has enough communication throughput performance required by high throughput applications such as 4K streaming, etc.

5 Conclusion

This paper has proposed and redesigned the CYPHONIC adapter to provide a sequential processing scheme. The new design stores an arriving order of incoming packets and can process the packets parallelly. Additionally, it can send processed packets according to the arrival order, considering the difference in processing time for each packet. As a result of the performance evaluation, we confirmed that the throughput was significantly improved for TCP and there were also no significant delays in processing.

DOI: 10.1587/comex.2023XBL0010
Received January 24, 2023
Accepted February 9, 2023
Publicized February 17, 2023
Copyedited May 1, 2023