Journal of Systemics, Cybernetics and Informatics (2023) 21(1), 18-25
https://doi.org/10.54808/JSCI.21.01.18

Extension mechanism of overlay network protocol to support digital authenticates

Kazushige MATAMA
Graduate School of Business Administration and Computer Science, Aichi Institute of Technology,
1247 Yachigusa, Yakusa, Toyota, Aichi 470-0392, Japan

Ren GOTO
Graduate School of Business Administration and Computer Science, Aichi Institute of Technology,
1247 Yachigusa, Yakusa, Toyota, Aichi 470-0392, Japan

Chihiro NISHIWAKI
Graduate School of Business Administration and Computer Science, Aichi Institute of Technology,
1247 Yachigusa, Yakusa, Toyota, Aichi 470-0392, Japan

Katsuhiro NAITO
Faculty of Information Science, Aichi Institute of Technology,
1247 Yachigusa, Yakusa, Toyota, Aichi 470-0392, Japan

ABSTRACT [1]

Zero-trust security is a new security model that has recently received much attention. Since the model protects all resources, continuous authentication and authorization of resources are mandatory. Many enterprises currently use cloud systems to manage their resources and provide service. On the other hand, IoT systems typically require cooperation service among IoT devices. As a solution for redundant routes and load on the cloud, a peer-to-peer type system is a good candidate. On the contrary, it requires zero-trust security because each device should guarantee security. Since the authors have proposed and developed CYber PHysical Overlay Network over Internet Communication (CYPHONIC) as a fundamental technology to realize zero-trust security, this paper introduces Public Key Infrastructure (PKI) into CYPHONIC. It proposes an extended device authentication scheme and a key exchange mechanism using digital certificates. According to the PKI mechanism, a certification authority authenticates the system and its authenticity of system, allowing communication with the correct communication partners. The proposed extension performs mutual authentication with digital certificates at the start of communication and secure encryption key exchange for communication between endpoints. We develop the proof of concept of the proposed scheme to confirm the adequacy of the extended mechanisms.

Keywords: IoT, Cloud services, Zero-trust security, Peer-to-Peer, Overlay network protocol, PKI

1. INTRODUCTION

Zero-trust security has attracted much attention in recent years. The model protects all resources rather than protecting the network, as in the past [1]. Therefore, continuous authentication and authorization of resources are mandatory regardless of the network location. Many enterprises use cloud systems to manage their resources, and many services rely on cloud systems [2]. On the other hand, Internet of Things (IoT) systems are also increasing with the rapid spread of embedded devices [3]. IoT systems typically require a cooperation service among IoT devices [5]. When a cloud-based system cooperates with IoT devices, redundant routes, load on the cloud due to data concentration, and the cloud system with a single point of failure become a big issue [4]. One of the solutions for these issues is a peer-to-peer type system. A secure peer-to-peer connection in different networks requires zero-trust security.

IoT devices typically use Internet Protocol (IP) technology to access the Internet. However, there are several issues on the Internet. The first is the NAPT (Network Address Port Translation) traversal, where NAPT routers block incoming packets from the Internet side [6]. The second one is the compatibility issue between IPv4 and IPv6 because each protocol version has different communication specifications. The third one is the session interruption due to the change of IP addresses during communication [7]. As to conventional research, some technologies have been proposed to solve these issues [8]–[10]. On the contrary, understanding the details of these technologies is not easy for users. Therefore, almost all users require a more fundamental framework to achieve zero-trust security.

We have proposed and developed CYber PHysical Overlay Network over Internet Communication (CYPHONIC) as a fundamental technology to realize zero-trust security [11]–[13]. CYPHONIC supports peer-to-peer type services for zero-trust security. It provides secure communications by authenticating user devices and encrypting communications in conjunction with the cloud. The initial prototype of CYPHONIC supported only password authentication for device authentication to the cloud. Since typical IoT devices should work without additional authentication settings by users, CYPHONIC requires another authentication mechanism instead of password authentication. In

[1] In preparing this paper, the final version of this paper was peer-reviewed by Hidekazu Suzuki, Ph.D. He is an associate professor at Meijo University, Japan. We would like to thank Dr. Hidekazu Suzuki for his valuable assistance in reviewing and editing this manuscript.

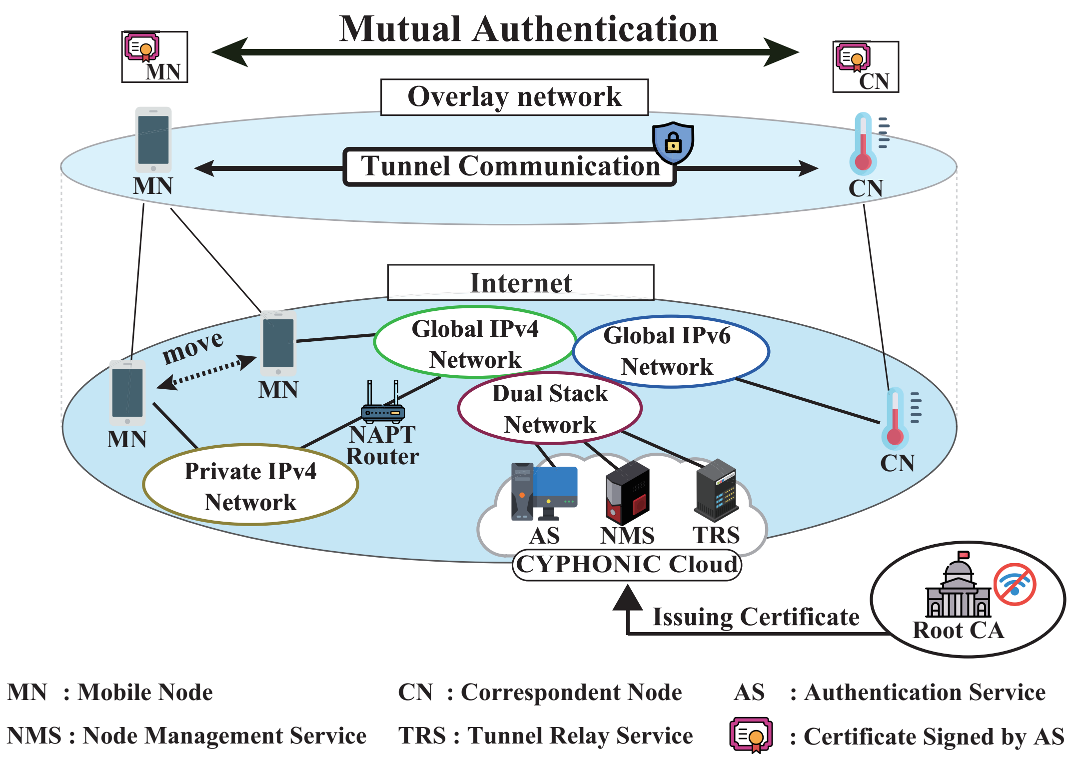

Fig. 1. Overview of CYPHONIC

addition, it relies on the trusting relationship between the cloud server and the IoT device to establish secure communication between devices. Therefore, it risks eavesdropping on the cloud system due to cyber attacks.

This paper introduces Public Key Infrastructure (PKI) into CYPHONIC and proposes an extended device authentication scheme and key exchange mechanism using digital certificates [14]. Secure Sockets Layer (SSL)/Transport Layer Security (TLS) communication uses a security infrastructure based on public key cryptography called PKI for authentication and encrypted communication [15]. A certification authority authenticates the system and authenticity of the system, allowing communication with the correct communication partners. Digital certificates can be used to authenticate without direct user intervention. They are easier to manage devices than password authentication because a digital certificate is issued for each device. Furthermore, mutual authentication with digital certificates at the start of communication between devices prevents reliance on cloud services. Additionally, the public key included in the digital certificate makes it more secure to share the End Key used for communication between the endpoints. We develop the proof of concept of the proposed scheme to confirm the adequacy of the extended mechanisms.

2. CYPHONIC

CYPHONIC achieves secure end-to-end communication by constructing a User Datagram Protocol (UDP) tunnel between CYPHONIC nodes over physical networks. Fig.1 shows an overview of CYPHONIC. It defines detailed procedures between cloud services and CYPHONIC nodes, which are end nodes. Cloud services consist of the following three types

- Authentication Service (AS)
  AS manages the device information of CYPHONIC nodes and performs authentication of CYPHONIC nodes. It distributes the Fully Qualified Domain Name (FQDN), the identifier of the CYPHONIC node, the virtual IP address, and the common key used for encrypted communication with NMS.
- Node Management Service (NMS)
  NMS manages network information of CYPHONIC nodes and controls the tunnel construction process between CYPHONIC nodes according to the network information.
- Tunnel Relay Service (TRS)
  TRS relays tunnel communication when direct communication is unavailable, such as between IPv4 and IPv6 or the packet blocking by NAPT routers.

CYPHONIC nodes have two major processing steps: processing before communication and processing to establish a communication tunnel between peer nodes. The pre-processing includes the authentication process and registration process.

Fig.2 shows the sequencing process of the conventional authentication process. In the authentication process, the CYPHONIC node performs login authentication to AS using SSL/TLS communication to show the authenticity of the CYPHONIC node to the cloud service. It performs the NMS registration process after the AS authentication process. In the registration process, the CYPHONIC node registers its network information through encrypted communication with NMS using

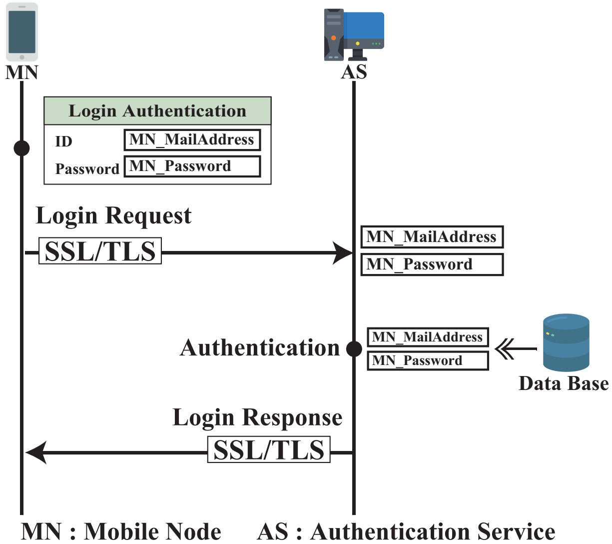

Fig. 2. Conventional login authentication process

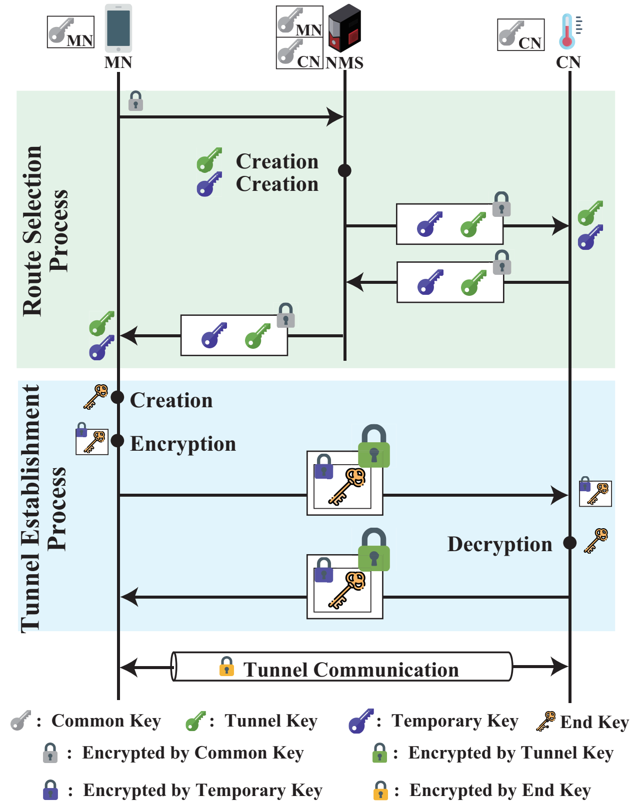

Fig. 3. Conventional communication establishment process

the common key distributed by AS.

The processes performed at the start of communication include route selection and tunnel establishment.

Fig.3 shows the sequencing process of the conventional communication establishment process. Mobile Node (MN) of the CYPHONIC node establishes tunnel communication according to the instructions of NMS when it tries to communicate with Correspondent Node (CN). MN receives instructions from NMS on the communication path to CN using the CN's FQDN in the route selection process. MN and CN receive two common keys to securely exchange the end key when MN and CN receive communication path instructions from NMS. NMS distributed two common keys: a common key (Temporary Key) for encrypting the end key and a common key (Tunnel Key) for encrypting the communication in the tunnel establishment process. When TRS relays the tunnel, the temporary key secures the end key. MN establishes a tunnel with CN after MN finishes the route selection process.

In the tunnel establishment process, MN generates an end key for communication with CN, encrypts the message with the tunnel key and sends it to CN. In this case, the end key is encrypted with the temporary key when the tunnel communication passes through TRS. CN decrypts the encrypted message received from the MN and extracts the end key. The CYPHONIC node establishes end-to-end communication by performing the above sequence. Both CYPHONIC nodes perform bidirectional communication after the communication establishment process. Since applications use virtual IP addresses, the communication session can be continuously available when the physical IP addresses change.

3. PROPOSED SYSTEM

This paper introduces PKI for the digital certificate authentication mechanism of CYPHONIC nodes. The initial prototype of CYPHONIC only supports password authentication for CYPHONIC nodes. Password authentication is a general authentication scheme for human operations. On the contrary, it is unsuitable for IoT devices because users may not operate them directly. Additionally, CYPHONIC nodes share an end key using a distributed key from NMS in the prototype implementation. Since the trust of both nodes relies on cloud service reliability, mutual authorization between CYPHONIC nodes is a countermeasure for attacks on the cloud service.

Current authentication schemes use public key cryptographic digital signatures for end-device authentication. The cloud service manages digital signatures according to the PKI. When a certification authority performs authentication, it performs verification using digital certificates. The digital certificate contains information about the certifier, the public key for encryption, and the issuer's digital signature. Therefore, this paper introduces PKI to CYPHONIC and proposes authentication and encryption using digital certificates. The proposed system introduces digital authentication using digital certificates and password authentication into the CYPHONIC authentication scheme. Using digital certificates eliminates the need to create your own highly secure passwords. Issuing digital certificates for each device realizes simple facilitating management. The proposed system also uses digital certificates to share end keys for end-to-end communication. Mutual authentication between the CYPHONIC nodes is possible by exchanging digital certificates when sharing the end key. Using the public key included in the digital certificate makes it possible to share the end key without the cloud system. As a result, the system can realize strong security.

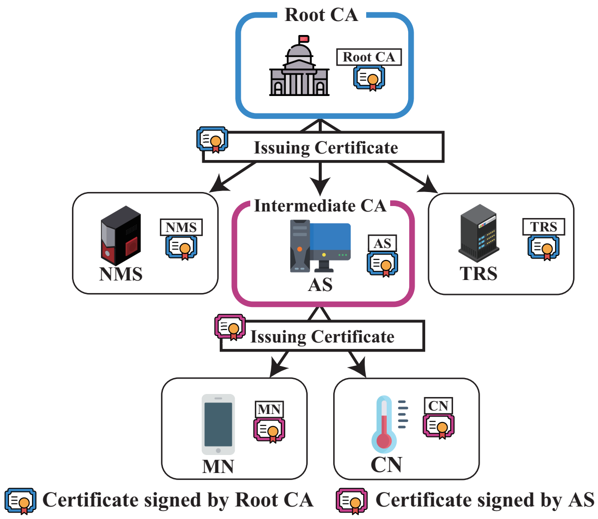

Fig. 4. PKI implementation model of CYPHONIC

3.1. Introduction of PKI

Fig.4 shows the proposed PKI deployment model. Since PKI requires root CA to issue certificates, we introduce our own root CA (Root CA) for CYPHONIC. We introduce server certificates for AS, NMS, and TRS based on the Root CA because Root CA should be stored offline. The proposed mechanism extends AS to manage certificates for CYPHONIC nodes, and AS operates as an intermediate certification authority. AS can issue digital certificates to the CYPHONIC node by the delegated function as a certification authority by Root CA. CYPHONIC node can show its authenticity by using a digital certificate issued by AS. Therefore, the proposed system can provide certificates for authentication at the start of service and during communication with the other nodes. Details of the changes made are described below.

- Root CA
  Root CA is a certification authority that issues digital certificates for CYPHONIC's cloud services. All elements using CYPHONIC should hold a root certificate issued by Root CA. The root certificate is a self-signed certificate used by Root CA to prove its authenticity and to verify each digital certificate's authenticity. Root CA issues server certificates to AS, NMS, and TRS. Root CA delegates the CA function only to AS. Additionally, Root CA delegates CA functions to AS to allow AS to operate as an intermediate certification authority.
- AS
  Since AS manages information on the device, it operates as an intermediate certification authority. AS can issue digital certificates to CYPHONIC nodes that have CYPHONIC accounts. The issuer's digital certificates verify the digital certificates of the CYPHONIC node, so digital certificates of Root CA and the intermediate certification authority, AS, are required.
- NMS
  NMS is a service that provides routing instructions at

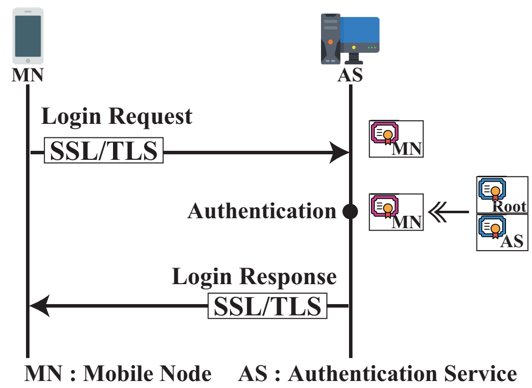

Fig. 5. Proposed login authentication process

  the start of communications between CYPHONIC nodes. It assists the CYPHONIC nodes in exchanging digital certificates for routing instructions.
- CYPHONIC Node
  CYPHONIC node uses the digital certificate issued by AS to authenticate with AS and share the common key used when communicating with the other CYPHONIC node. For this purpose, this paper implements a function for generating certificate requests. CYPHONIC node holds a root certificate issued by the root CA and an intermediate certificate issued to AS.

3.2. Signaling Extension

The proposed system extends the CYPHONIC signaling processes. The two extended signalings are the authentication process and the communication establishment process. CYPHONIC authenticates CYPHONIC nodes through the authentication process. The proposed system adds digital authentication to the authentication method. In digital authentication, AS issues a digital certificate to each CYPHONIC node. Then, the CYPHONIC nodes use the digital certificate for authentication. The authenticity of the CYPHONIC node is ensured through verification by the root certificate and by comparing the account information in the digital certificate with the registered information. The communication establishment process is performed when the CYPHONIC node starts communication with a peer CYPHONIC node. The communication establishment process involves route selection and tunnel establishment. The route selection process determines the communication path to the other node in cooperation with NMS. In conventional signaling, NMS distributes two common keys, the temporary key, and the tunnel key, once the communication path is determined. The extended signaling eliminates the temporary key and introduces the exchange of digital certificates with each other. The exchanged digital certificates are verified against each other using intermediate and root certificates. Both nodes verify authenticity mutually with the digital certificates. In addition, the digital certificate guarantees that the public key enables the secure public key exchange of the correct owner. Thus, public key cryptography enables end-to-end security measures without relying on the cloud. Details of the process to be extended are

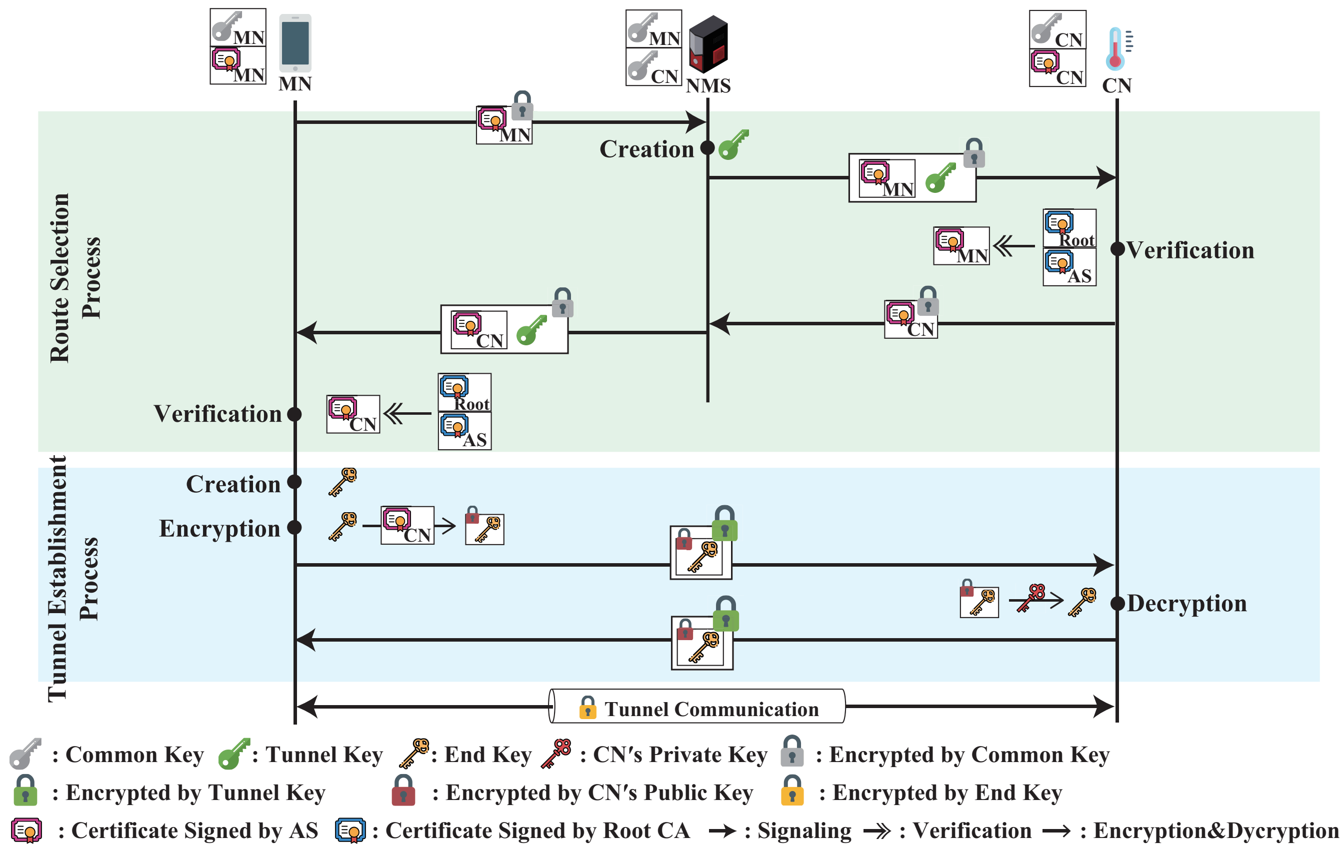

Fig. 6. Proposed communication establishment process

described below.

- Login authentication process
  Fig.5 shows a signaling diagram of the proposed login authentication process. The login authentication process works as follows.
  1) CYPHONIC node sends an authentication request to AS using a digital certificate issued by AS.
  2) AS verifies the digital certificate received from the CYPHONIC node with the intermediate certificate of AS and the root certificate of Root CA.
  3) AS accesses the database based on the information in the certificate and performs authentication by comparing it with the registered information.
- Communication establishment process
  Fig.6 shows the signaling process of the proposed communication establishment process. The communication establishment process involves two processes: route selection and tunnel construction. In the route selection process, mutual authentication is performed between the CYPHONIC nodes, and public key cryptography is used for more secure sharing in the tunnel construction process. Details of each process are described below.
  The route selection process works as follows.
  1) CYPHONIC node sends a packet containing the FQDN of the communication partner and its digital certificate to NMS.
  2) When NMS receives a communication selection request from MN, it determines a suitable communication route based on the network information of MN and CN.
  3) NMS sends the tunnel key and the digital certificate of MN to CN in the route instruction message.
  4) CN authenticates MN based on MN's certificate received from NMS.
  5) After authentication, CN's digital certificate is attached to the message and sent to NMS.
  6) NMS relays the message received from CN to MN.
  7) MN authenticates CN with the digital certificate from CN.

The tunnel establishment process works as follows.

1) MN generates the end key with CN.
2) MN encrypts the end key with the public key taken from CN's digital certificate.
3) MN encrypts the message with the tunnel key and sends it to CN.
4) CN combines the message received, including the tunnel key, to extract the encrypted end key.
5) CN decrypts the encrypted end key with CN's private key and extracts the end key.
6) CN notifies MN that it receives the message successfully.

3.3. Extension of Packet control functions

The proposed system exchanges digital certificates during the signaling of the tunnel establishment process. The size of the digital certificates exceeds the MTU size, which is the limit of the transmission size. CYPHONIC uses Transmission Control Protocol (TCP) for the authentication process and UDP communication for the registration and route-establishing process. Since the CYPHONIC node exchanges the digital certificates through UDP communication, the limitation of the MTU requires its splitting function of the digital certificates to convert it. Therefore, this paper extends the packet control function to provide fragmentation and retransmission processes to ensure sending of packets in segments. In conventional CYPHONIC, a proprietary header is added for communication. The conventional proprietary header contains identifiers that indicate the type of transaction or message and the original sender, and signaling processing is performed based on this information. The proposed system fragments data for communication because there is a possibility that the order of packets may be switched before reaching the destination or packets may be lost on the way. Therefore this paper adds a fragment header to manage the fragmented data. The fragmentation header is CYPHONIC's extended header for controlling fragmented packets. Fragmentation packets contain information on the length of the entire packet and the position of the split packet when it is reassembled. Details of each function are described below.

- Fragmentation process
  The fragmentation process provides for resizing packets exceeding the MTU size. Sending Node (SN) splits the data based on the MTU size. SN attaches the fragment header to the fragmented data and generates multiple packets. Receiving Node (RN) temporarily buffers the multiple packets received from the SN. RN checks whether all packets arrive in the buffer to reconstruct the original data. When RN finds all fragmented packets in the buffer, it reconstructs the data based on the information in the packets and replies with an acknowledgment packet. The fragmentation process makes it possible to send and receive data that exceeds the MTU value.
- Reconstruction process
  The retransmission process provides reliable delivery of packets. The fragmentation process splits data according to its size and generates multiple packets accordingly. Increasing the number of packets in communication causes the possibility of packet loss during the delivery process. The retransmission process prevents data from failing to reach RN by resending the lost packets. Therefore, it performs automatic repeat-request (ARQ) by confirming the packet delivery status at RN. As confirmation, RN also replies with an acknowledgment packet.

4. IMPLEMENTATION

We have developed the implementation of the authentication function using digital certificates and the end key sharing function on Linux OS. Since conventional CYPHONIC uses Golang to implement the functions, the proposed system also employs Golang to extend the functions. To introduce PKI into CYPHONIC, this paper used Open SSL, a library to support cryptographic algorithms. National Institute of Standards and Technology (NIST) recommends a key length of 2048 bits of RSA for the public key cryptographic algorithm and 128 bits of AES for the symmetric key cryptographic algorithm in its guidelines for cryptographic settings [16]. Therefore, the algorithms used in this paper also conform to these guidelines. Therefore, we will adopt these guidelines for the algorithms used in this paper. This paper used x.509v3 as the format for issuing digital certificates, with the public key used in digital certificates generated with a 2048-bit RSA public key.

The packet control process requires storing all the fragmented packets to reconstruct the original data. Therefore, RN stores the fragmented packets in an in-memory cache. The header information identifies fragmented packets identical to the data to be restored. Furthermore, the control process performs retransmission processing to ensure packet delivery. SN temporarily stores the sent packet in an in-memory cache because it may resend it again. The retransmission process continues at regular intervals until an acknowledgment packet arrives from RN.

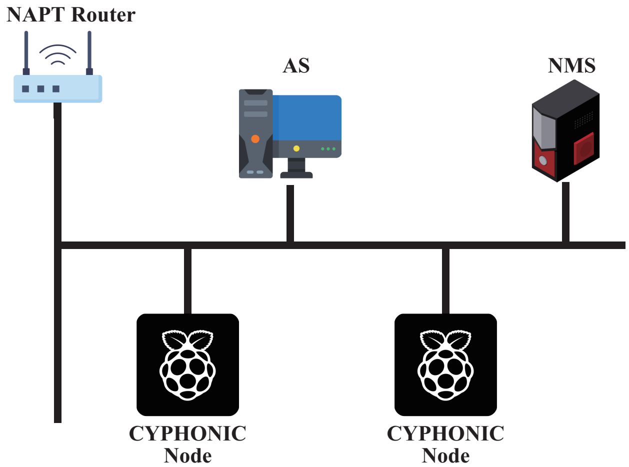

Fig. 7. Evaluation model

TABLE I
SPECIFICATIONS OF THE MEASURING DEVICES

CYPHONIC Cloud (AS,NMS)	
Machine	Virtual Machine
OS	Ubuntu 21.10
CPU	3.50GHz 2cores Intel(R) Core i9-11900K
Memory	2GB RAM
CYPHONIC Node	
Machine	Raspberry Pi 3 Model B
OS	Raspbian GNU/Linux 10.0 (Buster)
CPU	Quad Core 1.2GHz Broadcom BCM2837 64bit
Memory	1GB RAM

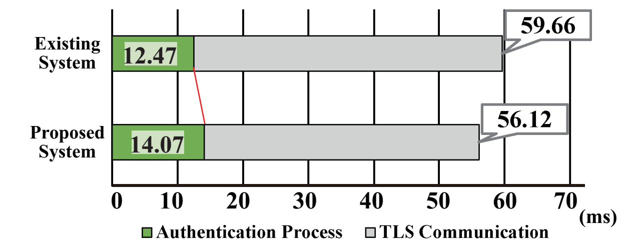

Fig. 8. Login authentication processing time measurement

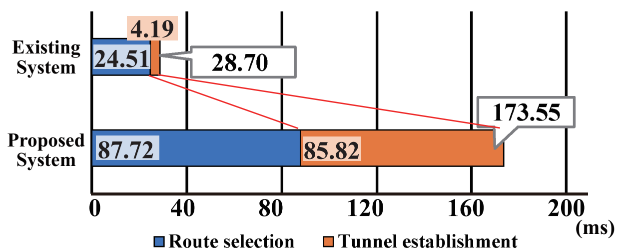

Fig. 9. Communication establishment processing time measurement

5. EVALUATION

Fig.7 shows the verification environment in this evaluation. In this verification, we prepared AS, NMS, and two CYPHONIC nodes on the same network. TableI shows the specifications of each service and device used in the verification. The cloud service used a virtual machine, and the CYPHONIC node used a Raspberry Pi 3.

This paper measured the processing overhead and communication performance associated with system expansion in the experimental evaluation. As the processing overhead, it evaluates the performance of the extended authentication and communication establishment processes. Fig.8 shows the measurement results of the authentication process. The measurement results for each process are averages of 30 trials. In the authentication process, we compared the conventional implementation of password authentication with the additional implementation of digital authentication.

The difference in simple processing shows that digital authentication has a higher overhead due to the additional verification of digital certificates. However, the total processing time was relatively unchanged. The reason is a large amount of time involved in SSL/TLS communication when the CYPHONIC node authenticates with AS.

We measured the processing time until the CYPHONIC node started communication in the communication establishment process. Fig.9 shows the measurement results of the communication establishment process. The total processing time of the conventional system is about 27.8 [ms]. In contrast, the proposed system incurred approximately 176 [ms]. The proposed system sends digital certificates attached to packets. Since the size of the digital certificate exceeds the packet size that can be sent at one time, the proposed system splits and reconstructs the packets. In addition, the public key cryptographic algorithm for encrypting the end key and the certificate authentication process increased

TABLE II
UDP THROUGHPUT PERFORMANCE

	Conventional System		Proposed System	
Traffic	Throughput	Jitter	Throughput	Jitter
10Mbps	10Mbps	0.63ms	10Mbps	0.71ms
20Mbps	20Mbps	0.71ms	20Mbps	0.74ms
30Mbps	29.6Mbps	0.42ms	29.6Mbps	0.37ms

the overhead.

We also evaluated the communication performance by measuring the communication throughput of the CYPHONIC nodes. TableII shows the communication performance of the conventional and proposed systems. We used the iperf tool to measure throughput between CYPHONIC nodes. The results were an average of 10 measurements. They showed that the throughput of both systems was almost the same. Since the proposed system only extends the processing before the start of communication, the proposed system confirmed that it does not affect communication performance. We found that the proposed system incurs overhead in the authentication process before the start of communication and the end key sharing process. The proposed system does not affect the performance after establishing the tunnel communication. Therefore, it has little impact on the operation as a service.

6. CONCLUSION

This paper has extended CYPHONIC to introduce digital certificates based on PKI and enhance security by updating the sharing method of encryption keys. Since AS works as an intermediate certification authority in PKI, it can issue a digital certificate to each CYPHONIC node. Each CYPHONIC node can realize a digital certificate-based authentication by authorizing digital certificates to each other. They also enhance the encryption key exchange mechanism by following the updated signaling process. The experimental verification confirmed that certificate-based authentication does not cause a significant overhead compared with password authentication. The evaluation results showed that the overhead is acceptable in the practical usage of CYPHONIC because it only temporarily increases during the tunnel establishment process.

ACKNOWLEDGMENT

This work is supported in part by Grant-in-Aid for Scientific Research (C)(21K11877), the Japan Society for the Promotion of Science (JSPS), and the Cooperative Research Project of the Research Institute of Electrical Communication, Tohoku University.

REFERENCES

[1] K. Charalampos, C Fabrizio, T. Dan, R Nathan,B. Elisa, “NEUTRON: A Graph-Based Pipeline for Zero-Trust Network Architectures,” Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy, April 2022.

[2] N. L. Singh, Akriti, P. Jaya, P. Roli, B. Sumitra, K. Sarvesh, “Security, Privacy Issues and Challenges In Cloud Computing: A Survey,” Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, March 2016.

[3] A. Mariam and T. Radwan, "Enabling Security as a Service for IoT Emerging Technologies: A Survey," The 7th Annual International Conference on Arab Women in Computing in Conjunction with the 2nd Forum of Women in Research, November 2021.

[4] Q. Lin, K. Hsieh, Y. Dang, H. Zhang, K. Sui, Y. Xu, J. Lou, C. Li, Y. Wu, R. Yao, M. Chintalapati, D. Zhang, "Predicting Node Failure in Cloud Service Systems," Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, November 2018.

[5] B. Sumathi, V. Hemalata, M. Raja Kumar, "Smart Home Technologies: A Preliminary Review," Proceedings of the 6th International Conference on Information Technology: IoT and Smart City, December 2018.

[6] H. Kavalionak, A. H. Payberah, A. Montresor, J. Dowling, "NAT-Cloud: cloud-assisted NAT-traversal service," Proceedings of the 31st Annual ACM Symposium on Applied Computing, April 2016.

[7] Á. Leiter, N.Galambosi, L. Bokor, "An Evolution of Proxy Mobile IPv6 to the Cloud," Proceedings of the 19th ACM International Symposium on Mobility Management and Wireless Access, November 2021.

[8] N. V. Keizer, and O. Ascigil and I. Psaras, and G. Pavlou, "Rewarding Relays for Decentralised NAT Traversal Using Smart Contracts," Proceedings of the Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, October 2020.

[9] Akilandeswari, D. and Rabara, S. Albert and Premila Bai, T. Daisy, "ECC-Enabled Translation Mechanism for Intercommunication Between IPv4 and IPv6," 2018 International Conference on Communication and Signal Processing (ICCSP), April 2018.

[10] Raza, Syed M. and P. Donghan and P. Yongdeuk and L. Kangwoo and C. Hyunseung, "Dynamic Load Balancing of Local Mobility Anchors in Software Defined Networking Based Proxy Mobile IPv6," Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication, January 2016.

[11] R. Goto, T. Yoshikawa, H. Komura, K. Matama, C. Nishiwaki, K. Naito, "Design and Basic Evaluation of Virtual IPv4 Based CYPHONIC Adapter," Proceedings of the 13th International Multi-Conference on Complexity, Informatics and Cybernetics: IMCIC 2022, March 2022.

[12] 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 40th International Conference on Consumer Electronics (ICCE), January 2022.

[13] T. Yoshikawa, H. Komura, R. Goto, K. Matama, C. Nishiwaki, and K. Naito, "Demonstration of video conferencing tool with overlay network protocol," 2022 IEEE 19th Consumer Communications Networking Conference (CCNC), January 2022.

[14] Sharon Boeyen and Stefan Santesson and Tim Polk and Russ Housley and Stephen Farrell and David Cooper, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," 10.17487/RFC5280, May 2008.

[15] Xia, Wei and Zhang, Qiyu and He, Xin and Wang, Wei and Li, Zhen and Xiong, Gang, "After Everything is Connected: A Client Certificate-Oriented Perspective of IoT Device Security Analysis," 2021 The 9th International Conference on Information Technology: IoT and Smart City, April 2022.

[16] E.Barker, Q.Dang, "Recommendation for Key Management, Part 3: Application-Specific Key Management Guidance," NIST Special Publications, January 2015.