Feasibility Verification of Access Control System for Telecommuting by Users Reliability Calculation

Atsushi Shinoda
Graduate School of Informatics
Nagoya University
Nagoya, Japan
email: shinoda@net.itc.nagoya-u.ac.jp

Hirokazu Hasegawa
Center for Strategic Cyber Resilience R&D
National Institute of Informatics
Tokyo, Japan
email: hasegawa@nii.ac.jp

Hajime Shimada
Information Technology Center
Nagoya University
Nagoya, Japan
email: shimada@itc.nagoya-u.ac.jp

Yukiko Yamaguchi
Information Technology Center
Nagoya University
Nagoya, Japan
email: yamaguchi@itc.nagoya-u.ac.jp

Hiroki Takakura
Center for Strategic Cyber Resilience R&D
National Institute of Informatics
Tokyo, Japan
email: takakura@nii.ac.jp

***Abstract*—Nowadays, telecommuting, in which users connect to a corporate network from remote locations, such as their homes, is increasing as a measure to prevent COVID-19 spread. However, telecommuting exposes companies to information security risks by allowing users to connect terminals from their home that is out of control. Further security enhancements are required for ensuring secure telecommuting, but they easily cause trade-off issues between security and business efficiency that the administrators have to solve. As a solution to this problem, we have proposed an access control system to minimize the loss of business efficiency while enhancing security. The system calculates the reliability of each connected user and implements network access control. This access control allows connection to many resources for business efficiency if the user's reliability is high, and minimizes the number of resources available for reducing risks if the user's reliability is low. This paper confirmed the feasibility of implementing the system to calculate reliability from realistic indicators and perform network access control using pseudo-corporate network.**

***Index Terms*—Access Control, ACL, SDN, Network Latency, Telecommuting, User Reliability**

I. Introduction

Nowadays, telecommuting, in which users connect to a corporate network from remote locations, such as their homes, is increasing as a measure to prevent COVID-19 spread and due to a development of information technology. It increases the number of work style options and increases business efficiency. However, for corporate networks, remote networks and terminals that are difficult to control by corporate administrators become a risk because their security is not guaranteed compared to terminals at the intranet. There is a need to enhance the security of corporate networks for telecommuting communications, but security enhancement measures often decrease business efficiency.

Therefore, we have proposed a solution to this problem: an access control system that enhances security but does not decrease business efficiency as much as possible [1] [2]. The proposed system calculates the reliability of each Virtual Private Network (VPN) connected user and determines which resources the user can connect to based on the importance of the resources. However, in the previous research, we have only proposed the mechanism of this system and verified the access control based on the pre-defined reliability indicators. We have not verified the feasibility of implementing the system that calculates reliability based on realistic indicators and its access control.

In this paper, we implemented the proposed system and verified how the implementation would affect the corporate network and remote connections. The results of the network latency evaluation in the intranet data transfer process during the remote connection confirmed that the intranet communication was not affected. In addition, the results of the experience degradation evaluation of the remote connection users showed that all connections to the corporate network were less than 1.0 second. However, connections to resources varied from 3.2 seconds at low load to 26.6 seconds with simultaneous remote connections and high load on the intranet.

The rest of this paper is organized as follows: Section II describes related work, Section III outlines the proposed system, Section IV describes the implementation, Section V describes the verification experiments, and Section VI provides a conclusion and future work.

II. Related Work

By registering Access Control List (ACL) in network equipment, such as routers, network access control can be implemented and corporate network can be more secure. Smetters et al. have conducted an extensive and long-term research on how access control with ACL can enhance security [3].

There is also research on using Software Defined Networking (SDN) to dynamically generate ACL to further enhance security [4]. While this research is only for the targeting of IoT devices, SDN is used as an IDS for network access control. We have to make high frequency ACL changes for dynamic access control in the proposed system. Therefore, we also use SDN as one of the easier ways to implement it. However, it can be assumed that ACL is normally configured on the network equipment, and management methods also exist. Liu et al. proposed ACL management method using optimization algorithm, it can make management easier by reducing redundancy, but it is not suitable for rapid ACL changes [5].

Dynamic control with SDN is expected to affect greater load on the network, so the impact on the network should be considered. There have been several research on the impact of SDN control in networks. Iqbal et al. proposed an analytical model for the end-to-end communication latency caused by centralized control using SDN, based on experimental measurements in a virtual environment and testbeds across the US and nine countries [6]. Llopis et al. proposed minimizing critical communication, such as remote surgery, latency reduction method in IoT communications by routing and redirecting to the shortest path using SDN [7]. This two research focus mainly on routing using SDN, whereas proposed system in this paper differs by focusing on network access control. Due to the fact that very few research have used SDN for network access control and measured speed, this paper cannot set a value as a baseline.

The proposed system in this paper uses user reliability to enhance security. For a concept similar to the user reliability, there are two research about scaling the self-report measure of user's security intentions [8] or attitudes [9]. As self-reported data is not accurate and dynamic, proposed system in this paper uses numerical data that can be obtained by automatically as a reliability indicator to calculate user reliability. In addition, there is research about investigation on Chief Information Security Officers' (CISO) awareness of Human-Centered security in corporation. Hielscher et al. indicate that many CISOs may not have enough knowledge about the latest academic Human-Centered security [10]. The system proposed in this paper to automatically determine the user reliability will be more required in the situation where the administrators (such as CISOs) do not have enough knowledge. Research also exist that focus on user's security awareness. While this paper discusses a method to measure user's security awareness, Masssoth et al. proposed learning method that uses AI to improve user's security knowledge and awareness [11].

III. Access Control System Using User Reliability

We explain about details of proposed system and user reliability calculation.

A. Overview of the Proposed System

It is difficult for companies to manage the networks and terminals which connected from outside of the companies. It is a risk to the corporate network because telecommuting communications can become a bridgehead. Therefore, it will be necessary to enhance security (e.g., strictly access control), but this is likely to sacrifice business efficiency. There is a need for a method that balances business efficiency and enhanced security.

Thus, we have proposed an access control system that aims to enhance security while minimizing the loss of business efficiency [1] [2]. This proposed system calculates the reliability of users who connect to the corporate network from outside for telecommuting and determines the user's accessible resources. The user's reliability is calculated based on an index that indicates the user's security awareness, and the importance of each resource is used to determine which resources to allow the user to connect to.

Users with high reliability are granted access to a wide range of resources. In contrast, users with low reliability are granted access to the minimum necessary resources. Furthermore, proposed system has a function that allows users to apply for access permission in case a user with low reliability needs to access a resource to which he/she has been denied access. If approved by the administrators, the user can connect to the resource temporarily for a certain period of time. It also has a dynamic access control function that periodically recalculates reliability to flexibly cope with time-changing network conditions.

B. User Reliability Indicators

The calculation of user reliability is based on indicators of user's awareness of security enhancement. Useable indicators for user reliability calculation are listed below.

- Security Training
  This reliability indicator can have two values, the progress rate of security training courses and test result scores during security training course.
- Incidents History
  This is the number of incidents that the user has caused in the past, such as being the target of a cyber attack that resulted in an information leakage.
- Security Surprise Test
  This is a result of unannounced surprised tests, such as a pseudo malicious e-mail that includes web beacon URL or attachment file includes beacon. If the user opens URL or attachment file, the score becomes bad. In addition, whether the user reported such risky behavior to the administrators comes to be another reliability indicator.
- Result of URL Filtering Detection
  This reliability indicator is the number of a user's attempts to follow malicious or suspicious URLs detected and prevented by the corporate Firewall or Unified Threat Management (UTM).
- Other Reliability Indicators
  Depending on the network environment, there are many other possible indicators, e.g., 'Windows security logs', 'Whether security updates are applied to installed programs', 'Number of spam e-mails received by users', and 'IP address of the telecommuting user's remote network'.

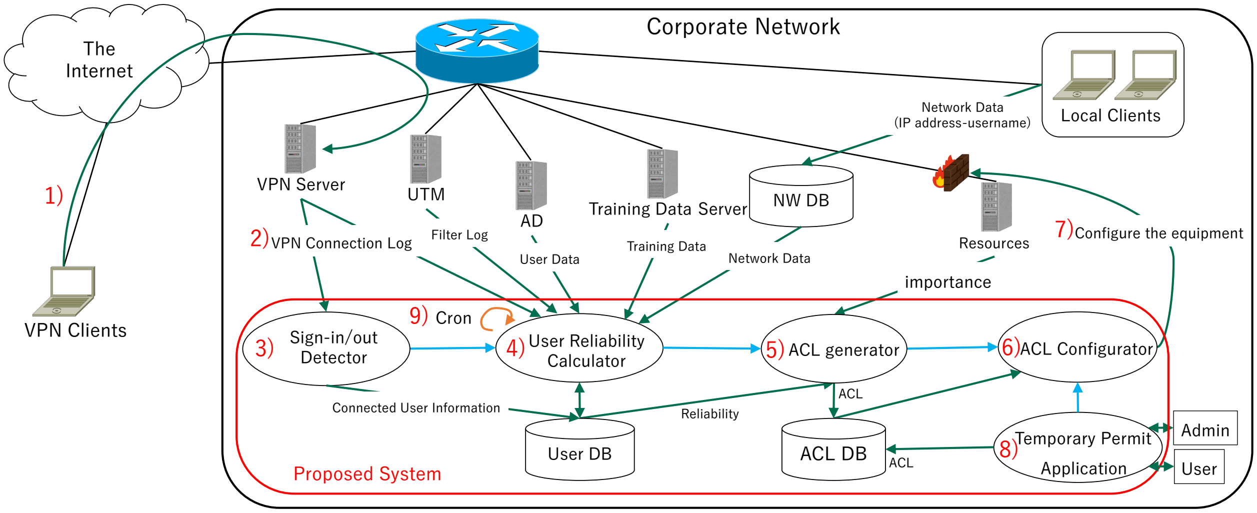


Fig. 1: Architecture of the Proposed System.

C. Calculation of User Reliability

The procedure of the reliability calculation based on each reliability indicator is shown as follows. First, for each reliability indicator, the data is standardized for all users, so that the value becomes relative and can be calculated by addition. Then, the reliability is calculated by adjusting a parameter to set the weight of each standardized reliability indicator. Parameter adjustment can be determined using statistical approaches, such as machine learning, or by dividing reliability indicators into some categories to reduce the number of parameters. Thus, administrator's effort for parameter adjustment may be reduced.

D. Generation of ACL

The proposed system determines the communication permission between the user's terminal and the server storing the important resources. We assume the importance of a resource can be determined by setting the level in advance by administrator or creator (such as setting level of confidentiality), or by using resource importance estimation methods, such as our previous research [12].

For the generation of ACL using network access control, it is necessary to determine the relationship between user reliability, resource importance, and stabilization constant(=threshold). As an example, Formula (1) is access control condition that grants permission to connect.

$$Reliability - Importance > StabilizationConstant \quad (1)$$

The stabilization constant is adjustable as a parameter and can be changed depending on company's security situation. For example, if threat intelligence indicates possible threat detection, such as the company is being targeted, or if a temporary increase of malicious communications to honeypots, or an increase of the number of malicious e-mails, we can increase the stabilization constant to make the company more secure. In contrast, if the situation subsides, the stabilization constant can be decreased to allow access to a wider range of areas to improve business efficiency. Also, by dividing the stabilizing constants by department, the number of parameters to be adjusted will increase, but it will be possible to create ACL that better fit the information being handled.

E. Architecture of the Proposed System

Fig. 1 shows the architecture of the proposed system and the corporate network assumed to apply the system. The description of each function in the processing procedure is as follows.

1) A VPN connection is made to the corporate network from a remote location for telecommuting.
2) The VPN server outputs the connection information such as the assigned IP address and username as a log to the proposed system at the time of connection.
3) Sign-in/out Detector detects the start/exit of the user's VPN connection from the received log and stores the connected user information including VPN connection ID, username, and assign IP address to the User Database (DB).
4) The User Reliability Calculator obtains several kinds of data from Active Directory (AD) and other servers and calculates the user's reliability based on the data. The calculated reliability is sent to the User DB.
5) ACL Generator generates ACL based on user's reliability and resource importance, and stores them in the ACL DB.
6) ACL Configurator receives the ACL stored in the ACL DB and passes them to network equipment.
7) Network equipment implements network access control based on the ACL.
8) The application for temporary permission also generates ACL based on the application and approval, and activates the ACL Configurator.
9) In addition, in order to flexibly cope with time-changing conditions, the ACL Generator is periodically invoked

TABLE I: Details About Network Elements.

Function	Element	Details
Router	Router and VPN Server	NEC IX2310
AD	OS	Windows Server 2019
Proposed System	OS	Rocky Linux 8.8
	SDN Protocol	OpenFlow 1.3
	SDN Controller	Ryu 4.3.4
	Programing	Python 3.6.8
	DB	SQLite 3.26.0
	Log Server	Rsyslog
SDN Switch	OS	Ubuntu 22.10
	SDN Switch	Open vSwitch 3.0.0
Resource Server	Host Machine OS	Windows 10 Professional
	Guest Machine OS	Rocky Linux 8.6
	File Sharing Protocol	SMB
Local Clients	OS	Windows 10 Professional
VPN Clients	OS	Windows 10 Professional
NW Info Server	OS	Rocky Linux 8.6
	File Sharing Protocol	SMB

to review the user's reliability and dynamically perform access control.

IV. PROPOSED SYSTEM IMPLEMENTATION

We implemented the access control system described in Section III into a pseudo-corporate network and verified it.

Fig. 2 shows the organization of the pseudo-corporate network in which the proposed system is implemented. Table I shows the detail breakdowns of elements in Fig. 2. In this implementation, unless otherwise noted, all processing is done by programs written in Python.

In this implementation, it is assumed that the network configuration information server (NW Info Server) combines several server functions, such as NW DB and Security Training Server. Therefore, NW Info Server also keeps information related combined functions such as some reliability indicators, client IP address and username correspondence table, and resource importance information. It is also assumed that one server was considered one resource.

The elements shown in Fig. 1 of the proposed system in this implementation are listed below.

A. Sign-in/out Detector

The VPN Server sends the logs using Syslog, and Rsyslog in the Sign-in/out Detector receives them. When received the log, Sign-in/out Detector is activated and extracts connection information (VPN connection number, VPN client's assigned IP addresses and username) from the log, and stores it in the User DB.

B. User Reliability Calculator

Indicators which we implemented for user reliability and their obtaining methods are shown below.

- Progress Rate of Security Training Courses, Test Result Scores During Security Training Course: These are stored in the NW Info Server as an Excel file, and User Reliability Calculator obtains it by Server Message Block Protocol (SMB).
- Incident History: It is associated with each user's information in AD, and User Reliability Calculator obtains it by LDAP.
- Result and Response of Security Surprise Test: These are stored in the NW info server as an Excel file, and User Reliability Calculator obtains it by SMB.
- Result of URL Filtering Detection: URL filtering is implemented by the Router function, and the log of IP address is obtained by Rsyslog. The table of correspondence between IP addresses and username of local clients is placed in the NW Info Server as an Excel file, and User Reliability Calculator obtains it by SMB.

We classified into the following categories. Some indicators classified into multiple categories.

- User Carelessness: Result of Security Surprise Test, Result of URL Filtering Detection, Incident History
- User Awareness of Efforts to Secure: Progress Rate of Security Training Courses, Response of Security Surprise Test
- User Skill Level: Test Result Scores During Security Training Course, Result of Security Surprise Test

The User Reliability Calculator obtains each reliability indicator for all users, standardizes them, and calculates user's reliability. User u's reliability $R(u)$ is as follows (2).

$$R(u) = \frac{1}{|\mathbb{C}|}\sum_{c\in\mathbb{C}}\frac{1}{|c|}\sum_{i\in c}(-1)^{K_i}\times v_i(u) \tag{2}$$

Here,
$\mathbb{C}$: set of categories,
c: category, set of indicators,
i: indicator,
K_i: attribute determined by indicator i,
$v_i(u)$: user u's reliability value of indicator i.

For indicators where higher values indicate less user reliable, '-1' is multiplied in the calculation. Therefore, K_i becomes 0 in case of indicators about 'Progress Rate of Security Training Courses', 'Test Result Scores During Security Training Course', 'Response of Security Surprise Test'. In contrast, K_i becomes 1 in case of indicators about 'Incident History', 'Result of Security Surprise Test', 'Result of URL Filtering Detection'.

Note that if a large number of VPN is connected at the same time, the User Reliability Calculator and ACL Generator will be started again for each connection, which spend large times. Therefore, after the Sign-in/out Detector detects a connection and stores it in the User DB, if more VPN connections are connected, the ACL Generator is not activating in the middle of the process, but only once at the end. The periodic execution is defined by Cron and is executed every minute.

C. ACL Generator

This function generates ACL for each connected VPN user to access resources. The intranet utilizes default deny policy so that we have to set allow ACL to access resources. A permission rule is generated based on the generation condition

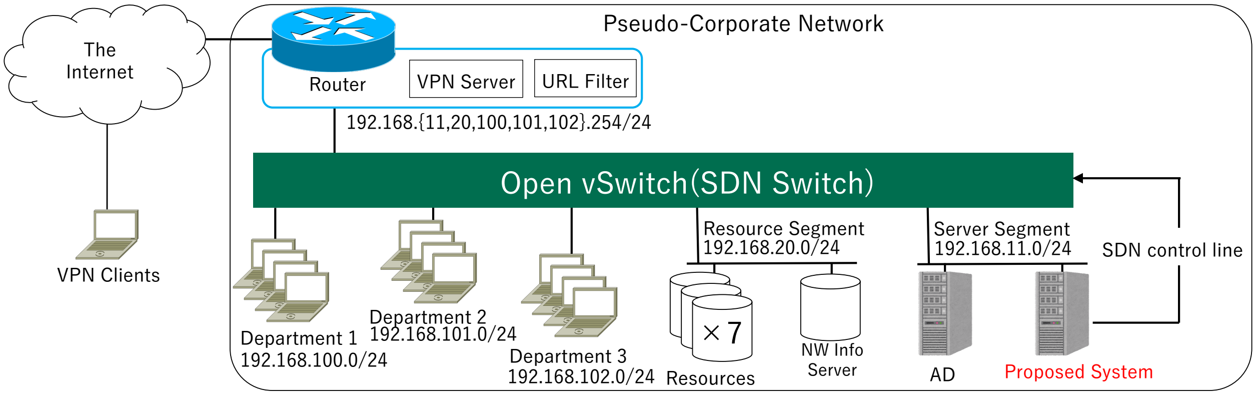


Fig. 2: Experimental Network.

Formula (1). In this implementation, Stabilization Constant value is '0'. The information of resources importance is already specified individually for each resource server, and is placed in the NW Info Server as an Excel file, which is obtained by ACL Generator via SMB.

D. ACL Configurator

The ACL Configurator obtains ACL to be configured from the ACL DB. The ACL Configurator implements Ryu [13] as an SDN (Protocol: OpenFlow) controller and configures the received ACL to the SDN switch by the firewall function of Ryu. In this implementation, access control from the local client to the resource server is performed by SDN, but the ACL are prepared in advance, so they are not affected even if the ACL are updated by a VPN connection.

V. Feasibility Verification Experiment

The proposed system enhances the security of corporate network by providing access control based on the user reliability for telecommuting. However, it is not desirable that the introduction of this system affects connection to and use of the corporate network. Therefore, experiments were conducted on the availability when a large number of remote users are connected to corporate network at the same time or when the corporate network is busy.

A. Overview of the Experiment

We measured the time required for a VPN connection and the time until the resource can be accessed (SMB connection) in the experimental environment. Since the ACL generated change with each additional connection, multiple VPN connections are measured. There are two possible patterns for these multiple connections:

- **Sequential** connections: the number of VPN-connected clients is increased one by one.
- **Simultaneous** connections: all VPN-connected clients are connected at the same time.

In addition, since the load on the SDN and resource server may vary depending on whether the local client is communicating, measurements will be taken for two patterns:

- **None**: no communication from the local client to the resource server.
- **Heavy**: an extremely large amount of communications are transported from the local client to the resource server.

The measurement on the VPN Clients is controlled by PowerShell Remoting and using Measure-Command. The load **Heavy** means sending large PING packets (51.6KB) every 1 second by 5 local clients to resource servers and receiving 600MB file via SMB by 7 clients from resource servers.

For each reliability indicator, the data was generated by random number generation after adjusting the generation range so that the ratio of high:medium:low reliable users is 2:6:2. All measurements were taken three times, and the values in the table are averages of the three.

B. Results of the Feasibility Verification Experiment

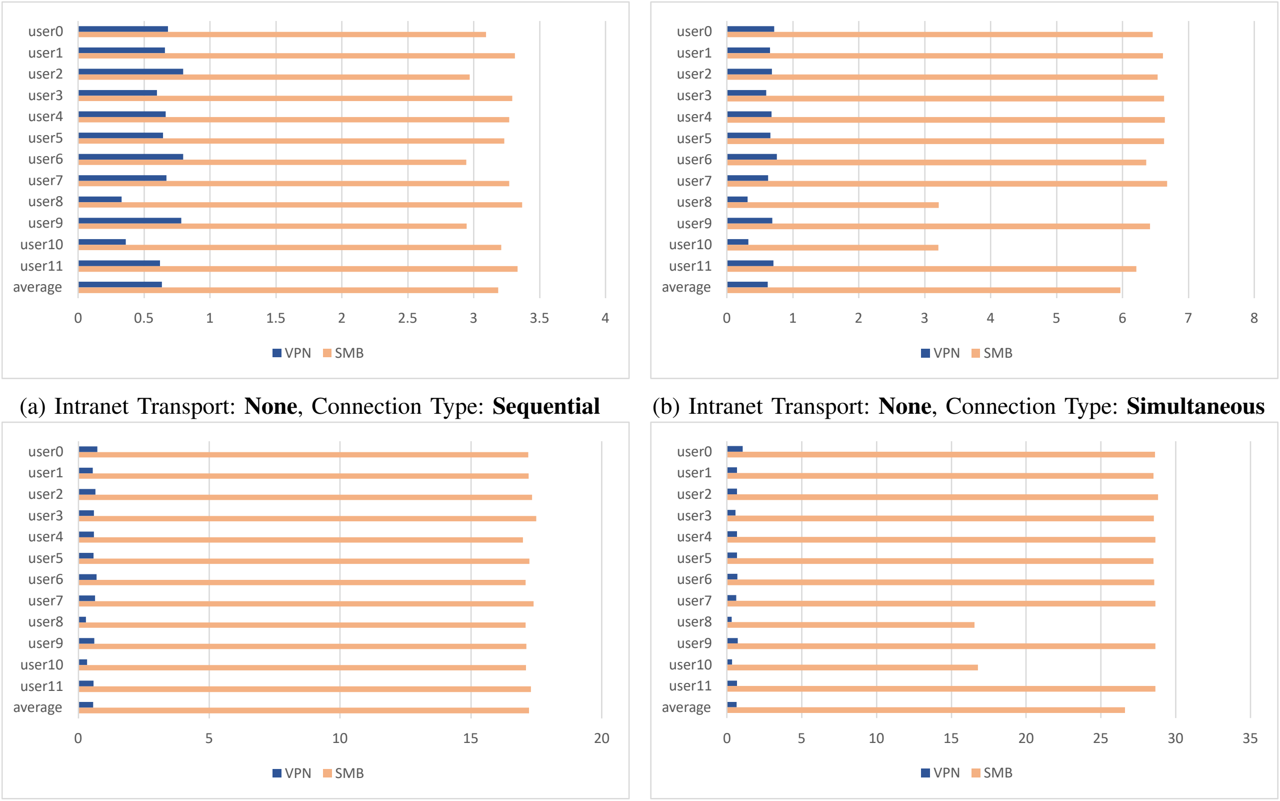
Measured VPN connection times and SMB connection times are shown in Fig. 3.

In the pattern of Transport: **Heavy**, Connection Type: **Simultaneous**, the SMB can not connect to because of too long waiting time, so trying to SMB connection is waiting until PING can be accepted (Note: the measured time at the pattern (**Heavy**, **Simultaneous**) in the Fig. 3d is the sum of the PING waiting time and SMB connection time).

C. Consideration

As shown in Fig. 3, the reason why there was no significant difference in the VPN connection was because the system's reliability calculation process was not started up yet.

The difference in SMB time for the **Sequential** was no more than 1 second, and the SMB connection time did not depend on the number of clients. In the **Simultaneous**, some clients (ID: user8, user10) took the same amount of time as in the **Sequential**, but other clients took twice as long. This was due to the processing of simultaneous VPN connections. Even though PowerShell was used to initiate the batch connection, there was slight difference between user8, user10 and other users. Based on such difference, the proposed system divided the connection into two sets. In fact, during the experiment, two VPN connections succeeded → ACL generator started →



(a) Intranet Transport: **None**, Connection Type: **Sequential**

(b) Intranet Transport: **None**, Connection Type: **Simultaneous**

(c) Intranet Transport: **Heavy**, Connection Type: **Sequential**

(d) Intranet Transport: **Heavy**, Connection Type: **Simultaneous**

Fig. 3: Results of VPN/SMB Connection Waiting Time(second).

TABLE II: Execution Time for Each Function(second).

Intranet Transport	None		Heavy	
VPN connection clients	0	12	0	12
get_file process	1.361	1.348	3.649	3.641
acl_config process	0.376	0.666	0.380	0.738
all processes	2.248	2.666	11.419	11.715

other VPN connections succeeded → ACL generator started, and so on.

When the network was under **heavy** load, it took some time for both the **Sequential** and **Simultaneous** to become available. This is because the ACL generator takes more than 10 seconds.

The time command was used to measure the execution time of each function of the proposed system under each situation, and the results are shown in Table II. *get_file process* is a function that User Reliability Calculator obtains a file from the NW Info Server, and it is used about 4 times. *acl_config process* is a function for the ACL Configurator to reflect the ACL database to the SDN switch. *all processes* is a series of processes performed by the User Reliability Calculator, ACL Generator, and ACL Configurator.

From the above, it can be assumed that the waiting time of **Sequential** under **heavy** load is due to *get_file process* being performed about 4 times, and the waiting time of **Simultaneous** is due to it being performed twice as many times by many clients. To solve this problem, it is necessary to shorten the access to some reliability indicators, so the value of the indicators should be calculated in advance and the process of passing it to the system side should be asynchronous, with high frequency.

Just in case, we checked the effect on the local client's communication using **heavy** load for pseudo-corporate network, but there was almost no change in the time taken for communication in both the **Sequential** and **Simultaneous**.

Impacts of the proposed system is limited because it is only applied immediately after the VPN connection is established. In this experiment, the network is under **heavy** load, which means that some of the bandwidth is overflowing, but it is difficult to imagine a network that is always in this state. However, if SMB communication is performed immediately after the VPN connection is established in a **Simultaneous**, sometimes an error raise and a delay is caused.

VI. Conclusion and Future Work

In this paper, we implemented access control system based on user reliability in the pseudo-corporate network environment. The verification using the network confirmed the feasibility of the proposed method. The proposed system focuses on the user's security awareness and the risk to the corporate network. The purpose of the proposed system is to balance security enhancement and business efficiency as much as possible, which easily cause trade-off issues in network access control.

The system was implemented on a pseudo-corporate network, and in the feasibility verification, waiting time for VPN connections and SMB connections were measured. The impact of the proposed system on VPN connections and the corporate network was limited. The impact of multiple connections was also limited if they were small scale. However, it was confirmed that if the network was under an abnormal load, the waiting time can not be ignored due to the time spent during the reliability calculation and ACL generation.

In future works, the exact validation of user reliability calculation has not been carried out in this paper. In addition, the reliability indicators should be based on as many indicators as possible so that the reliability can be calculated accurately. Furthermore, in order to be simplify and optimize the adjustment of parameters in calculating reliability, the proposed system should introduce a function that uses received feedback and automatically adjusts the parameters. With regard to the proposed system feasibility, the algorithm or system configuration should be reviewed to cope with the situation where significant waiting times were identified in the experiment.

Acknowledgment

This work was partially supported by JSPS KAKENHI Grant Numbers JP19K20268, JP23H03396, JP19K11961.

References

[1] H. Hasegawa and H. Takakura, “A dynamic access control system based on situations of users,” 7th International Conference on Information Systems Security and Privacy, pp. 653-660, 2021.

[2] A. Shinoda, H. Hasegawa, Y. Yamaguchi, H. Shimada, and H. Takakura, “Feasibility verification on impact of frequently access control update based on user reliability,” 9th International Conference on Information Systems Security and Privacy, Abstracts Track, 2023.

[3] D. K. Smetters, N. Good, “How users use access control,” 5th Symposium on Usable Privacy and Security, pp. 1-12, 2009.

[4] A. Hamza, H. H. Gharakheili, and V. Sivaraman, “Combining MUD policies with SDN for IoT intrusion detection,” 2018 Workshop on IoT Security and Privacy, pp. 1-7, 2018.

[5] A. X. Liu, E. Torng, and C. R. Meiners, “Compressing network access control lists,” IEEE Transactions on Parallel and Distributed Systems, Vol. 22, No. 12, pp. 1969-1977, 2011.

[6] A. Iqbal, U. Javed, S. Saleh, J. Kim, J. S. Alowibdi, and M. U. Ilyas, “Analytical modeling of end-to-end delay in openflow based networks,” IEEE Access, vol. 5, pp. 6859-6871, 2017.

[7] J. M. Llopis, J. Pieczerak, and T. Janaszka, “Minimizing latency of critical traffic through SDN,” 2016 IEEE International Conference on Networking, Architecture and Storage, pp. 1-6, 2016.

[8] S. Egelman, M. Harbach, E. Peer, “Behavior ever follows intention?: a validation of the security behavior intentions scale (SeBIS),” 2016 CHI Conference on Human Factors in Computing Systems, pp. 5257-5261, 2016.

[9] C. Faklaris, L. Dabbish, and J. I. Hong, “A self-report measure of end-user security attitudes (SA-6),” 15th Symposium on Usable Privacy and Security, pp. 61-77, 2019.

[10] J. Hielscher, U. Menges, S. Parkin, A. Kluge and M. Angela Sasse, “Employees Who Don’t Accept the Time Security Takes Are Not Aware Enough: The CISO View of Human-Centred Security,” 32nd USENIX Security Symposium, pp. 2311-2328, 2023.

[11] M. Masssoth, “Next Generation Artificial Intelligence-Based Learning Platform for Personalized Cybersecurity and IT Awareness Training: A Conceptual Study,” The Seventeenth International Conference on Sensor Technologies and Applications, pp 32-37,2023.

[12] Y. Zhou, H. Hasegawa, and H. Takakura, “A resource importance estimation method based on proximity of hierarchical position,” 5th International Conference on Information Science and Systems, pp. 83-89, 2022.

[13] RYU project team, “Ryubook 1.0 documentation,” 2014. https://osrg.github.io/ryu-book/en/html/index.html [retrieved: Oct, 2023]