Scientific Research Publishing

Communications and Network, 2026, 18(1), 1-9
https://www.scirp.org/journal/cn
ISSN Online: 1947-3826
ISSN Print: 1949-2421

IoT Security System Based on Software Defined Network with Manufacturer Usage Description

Takuma Sakamoto, Kenya Sato

Information and Computer Science, Graduate School of Science and Engineering, Doshisha University, Kyoto, Japan
Email: takuma.sakamoto@nislab.doshisha.ac.jp

How to cite this paper: Sakamoto, T. and Sato, K. (2026) IoT Security System Based on Software Defined Network with Manufacturer Usage Description. *Communications and Network*, **18**, 1-9.
https://doi.org/10.4236/cn.2026.181001

Received: November 3, 2025
Accepted: December 7, 2025
Published: December 10, 2025

Abstract

In recent years, the advancement of the Internet of Things (IoT) has significantly improved convenience; however, it has also increased security risks. Due to resource constraints, IoT devices have limited functionality, making it difficult to implement robust security measures. Additionally, users of home networks, one of the primary application areas of IoT, generally lack security expertise. Therefore, it is essential to develop a security system that meets the requirements of IoT devices while minimizing the burden on users. In this study, we propose a system that applies Manufacturer Usage Description (MUD) to home networks to define and control IoT device communication. This approach enables security measures to be implemented on IoT devices while reducing user burden. Furthermore, the proposed system monitors network traffic and isolates compromised devices upon detecting an attack, thereby minimizing potential damage. Through simulation based evaluations, we demonstrate that the proposed system effectively maintains security requirements.

Keywords

Home Network, Virtualization, Security

1. Introduction

In recent years, the advancement of the Internet of Things (IoT) has gained significant attention, leading to its rapid development. As a result, an increasing number of diverse devices are expected to connect to networks, further enhancing convenience [1]. The interconnection of various devices enables the collection and utilization of big data. Moreover, IoT devices are becoming more prevalent in local networks, such as home networks, expanding the scope of interconnectivity between devices [2]. These advancements are anticipated to improve the efficiency

and convenience of both daily life and industrial operations.

While the advancement of IoT enhances convenience, it also introduces increased security risks as devices that were previously not connected to networks become integrated into them [3]. Many IoT devices are developed without sufficient security considerations, making them vulnerable to weak passwords, inadequate privacy protection, and other security deficiencies [4]. Consequently, they are frequently targeted by malicious attackers. Compared to traditional computing devices such as PCs, IoT devices often lack sufficient computing power, network bandwidth, and memory capacity [5]. These hardware limitations, along with inherent software vulnerabilities, make it challenging to apply standard security measures, such as encryption, directly to IoT devices. Furthermore, users of home networks typically prioritize convenience over security and often lack awareness of the associated risks [6]. Given their limited security knowledge, configuring security settings independently is difficult for most users. Due to these vulnerabilities, if an attacker infiltrates a home network, they can freely access internal devices, increasing the risk of malware infections and potentially compromising the entire home network [7].

In this study, we define the communication requirements for each IoT device and apply the Manufacturer Usage Description (MUD) [8], which performs communication control, to home networks by leveraging Software Defined Network (SDN). This approach aims to reduce the burden on users while providing a lightweight security mechanism tailored to the communication requirements of individual devices. Additionally, our proposed system monitors network communications and, upon detecting an attack, isolates the compromised device from the network to prevent further damage and minimize security risks.

2. Related Work

The National Institute of Standards and Technology (NIST) conducted a study to evaluate the practicality and effectiveness of using MUD to enhance the security of IoT devices in small to medium-sized networks [9] [10]. The application of MUD with SDN enabled the flexible enforcement of communication policies. However, one major issue is that it does not address attacks originating from within the network, as devices can freely access each other. Once an intruder gains access, the security risk extends across the entire network. The number of devices connected within home networks is increasing, leading to heightened security risks [2]. Therefore, it is desirable to build a system that assumes the possibility of internal intrusions and minimizes damage in the event of an attack.

3. Proposed System

3.1. System Configuration

In this study, we apply MUD to a home network by utilizing OpenFlow [11], a representative protocol of SDN, and implement security measures through access control. In addition, in order to respond to attacks from inside the network, an

intrusion detection system is implemented on a router to monitor communications by relaying communications between devices.

The proposed system consists of the following components: IoT devices, a router, an intrusion detection system, an SDN controller, and an MUD file server. The details of the configuration are shown below.

3.1.1. IoT Devices

The IoT devices considered in this study are defined as devices that lack sufficient computational resources, such as CPU and memory, and therefore cannot implement direct security measures.

3.1.2. Intrusion Detection System

The intrusion detection system is capable of monitoring traffic from both external and internal networks. It performs traffic analysis at both volume and packet levels to detect intrusions by attackers and identify abnormal behaviors of IoT devices.

3.1.3. Router

The router serves as an intermediary for communication between IoT devices and also facilitates communication with external networks and servers. Additionally, the intrusion detection system is implemented within the router to monitor network traffic. The router is assumed to have sufficient resources to support these functions.

3.1.4. SDN Controller

The SDN controller retrieves MUD files that specify the communication requirements of each IoT device. It then enforces network communication control based on the policies defined in the MUD file.

3.1.5. MUD File Server

The MUD file server stores and manages MUD files that define the communication requirements for each IoT device.

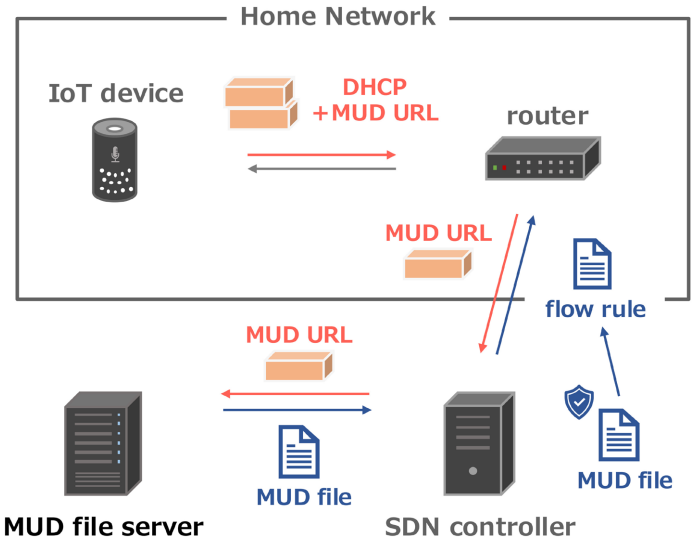


Figure 1. Application of MUD in home networks using SDN.

3.2. Application of MUD in Home Networks Using SDN

The flow of applying MUD is shown in Figure 1, with the details shown below.

1) Each IoT device issues its assigned MUD URL when it connects to the network.

2) The router forwards the MUD URL to the SDN controller.

3) The SDN controller retrieves the MUD file from the MUD file server using the received MUD URL.

4) The SDN controller verifies whether the retrieved MUD file was generated by the correct author, based on the certificate obtained from the MUD file server.

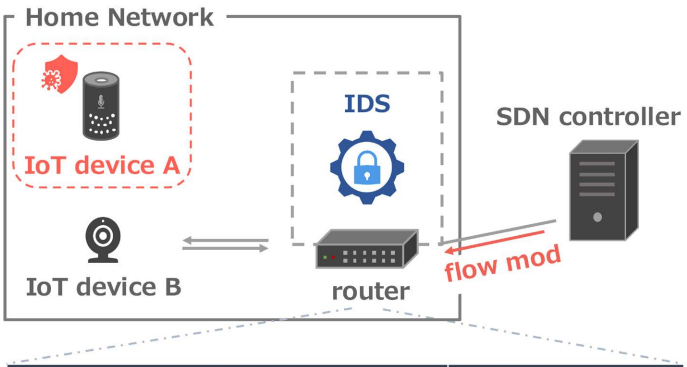
5) The SDN controller converts the communication requirements described in the retrieved MUD file from the access control entry format into flow rules and registers them in the flow table.

6) Communication is controlled according to the registered flow table.

This approach enables access control by configuring communication requirements such as protocols, port numbers, IP addresses, and domain names necessary for communication, thereby restricting IoT device communication to only the required transmissions. Since security measures are not directly implemented on IoT devices and user configuration is unnecessary, this method reduces user burden and mitigates security risks without being constrained by the resource limitations of the devices.

3.3. Attack Detection by Intrusion Detection System

In this study, network monitoring is performed using Snort, a signature based attack detection engine. Snort is a lightweight signature-based system that has been shown to be effective in environments where computational resources are limited and traffic volumes are not large [12]. In addition, the security community is continuously updating the rules and extending the functionality of the system to keep up with the latest threats.



match field	instruction
ip_src:172.16.0.A, ip_dst:172.16.0.B	drop
ip_src:172.16.0.B, ip_dst:172.16.0.C	send packet C
ip_src:172.16.0.B, ip_dst:172.16.0.A	drop

Figure 2. Flow table modification.

The proposed system implements Snort on a router that relays communications between devices and mediates communications from external networks, and analyzes network communications in terms of traffic volume and packet units. The system checks the packet headers, payload, etc., and matches them with a database. If a matching signature is found, the system determines that the packet is an attack or abnormal behavior. When an attack or anomaly is detected, an anomaly detection message is sent to the SDN controller and the user by email.

3.4. Minimization of Attack Damage

In this study, the minimization of attack damage is achieved by leveraging the flexible network configuration capabilities of SDN to modify flow tables. SDN adopts an architecture that separates the control and forwarding functions, unlike conventional network devices that integrate these functions. By designing and implementing the SDN controller, which serves as the control plane, network administrators can freely implement the necessary control functions. In the context of IoT applications, it is necessary to quickly adjust and optimize network communication speeds and data processing performance based on usage scenarios. Therefore, SDN is suggested to be suitable as a mechanism that keeps the overall functionality and performance of IoT systems always optimized [13].

The modification of the flow table is illustrated in **Figure 2**. When an attack originating from a compromised internal IoT device is detected, the SDN controller receives an anomaly detection message from the intrusion detection system. Upon receiving this message, the SDN controller updates the flow table to instruct all packets to and from the compromised device to be dropped. By completely isolating the compromised device from the network, this approach prevents further internal network damage and mitigates the risk of the device being exploited as a stepping stone for external attacks.

4. Evaluation

4.1. Evaluation Setup

The configuration of the implementation environment used in this study is illustrated in **Figure 3**. The implementation was carried out on Ubuntu, which was set up as a guest OS within VirtualBox running on Windows as the host OS. The IDS was implemented using Snort, a signature-based attack detection engine. The SDN controller was virtually created on Ubuntu using OpenDaylight, an SDN controller development platform. By utilizing OpenDaylight's RESTCONF interface, Mininet was integrated with the SDN controller and connected to Open vSwitch (OVS), enabling communication control based on the communication requirements defined in MUD files.

Furthermore, to evaluate the proposed system, a virtual home network environment was constructed using Mininet, a network simulator that supports OpenFlow. The network environment consisted of two hosts and two servers, with communication requests originating from a malicious server, a legitimate server,

and hosts. To simulate a scenario in which a compromised device initiates an attack within the network, an environment was set up where a host performs a Denial-of-Service (DoS) attack.

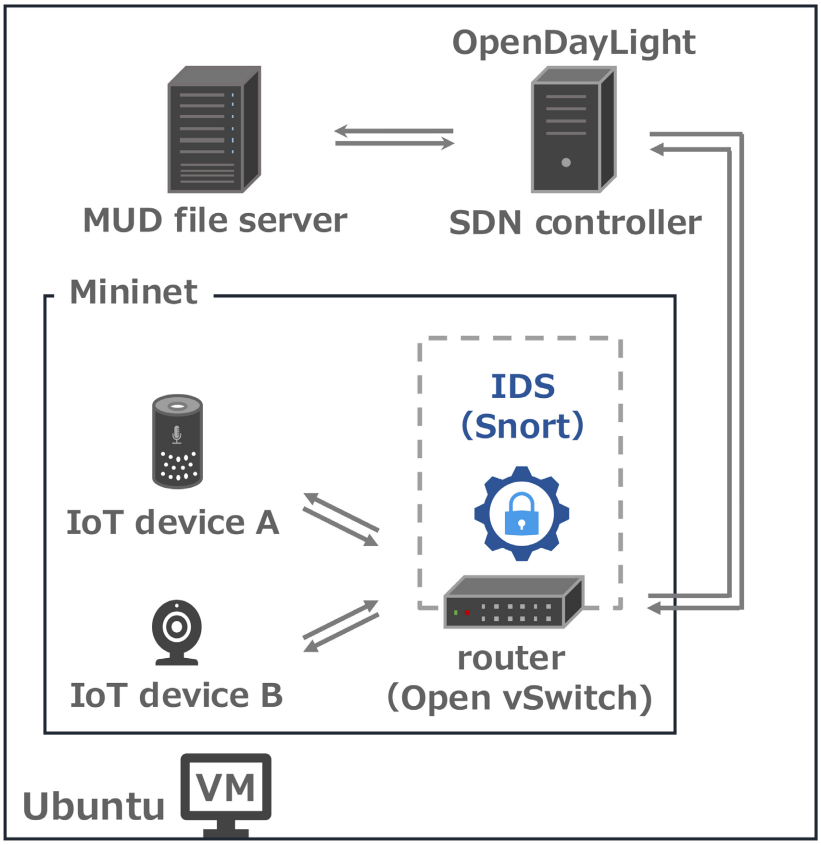


Figure 3. System configuration for implementation.

4.2. Evaluation Metrics

To evaluate the proposed system, we first verified whether the security measures were correctly applied. In this evaluation, we considered two types of abnormal communication:

1) Unauthorized communication from a server that is not permitted by the MUD file.

2) An attack originating from an internal network device.

As the attack method, we simulated a DoS attack using a SYN flood attack. Under these conditions, we examined whether permitted communications were correctly allowed, unauthorized communications were properly blocked, and whether attacks from internal devices were detected, leading to the isolation of the compromised device.

Additionally, to assess the effectiveness of the proposed system, we measured the Round Trip Time (RTT) between IoT devices. The average RTT was calculated over 10 communication attempts. As a baseline for comparison, we also measured the RTT in a scenario where no security measures were applied.

5. Result and Discussion

The results of communication between IoT devices, an authorized server, and an unauthorized server, as well as the flow table outcomes when an attack originates from an internal device, are shown in Figure 4. The results confirm that communication with the authorized server is permitted according to the rules defined in the MUD file, while communication with the unauthorized server is blocked. Furthermore, when an attack occurs from within the network, the system detects the anomaly, notifies the SDN controller, and generates a flow table that drops packets associated with the attacking device.

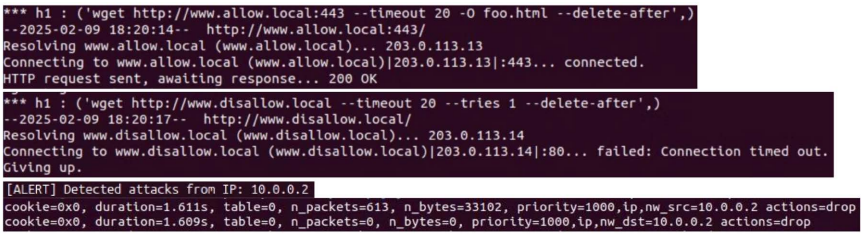


Figure 4. Communication to the authorized server (Top); communication to the unauthorized server (Middle); flow table for dropping packets upon anomaly detection (Bottom).

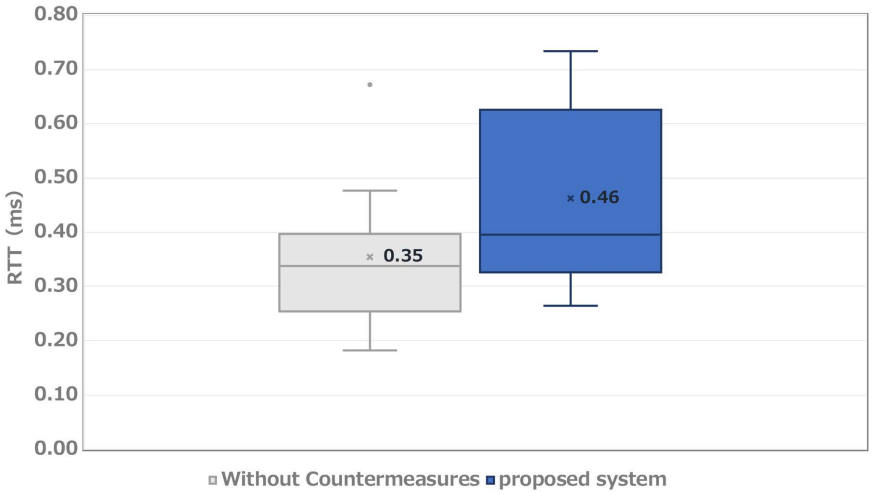


Figure 5. RTT in device-to-device communication.

Next, Figure 5 presents a comparison of inter device communication between the proposed system with security measures and a system without security measures. The results indicate that, compared to the system without security measures, the proposed system exhibits an increase in the average round-trip time by approximately 0.11 ms.

As shown in Figure 4, by applying MUD to home networks utilizing SDN, we demonstrated that IoT device communication can be restricted to only necessary traffic, thereby enforcing access control and reducing security risks. Additionally, the intrusion detection system successfully detects attacks originating from within

the home network and immediately updates the flow table, enabling the isolation of the attacking device. This isolation not only prevents the spread of damage within the network but also mitigates the risk of the compromised device being exploited as a means to launch external attacks.

Furthermore, Figure 5 indicates that the proposed system introduces some delay compared to an environment without security measures. This delay is attributed to the processing load involved in verifying whether incoming packets match the flow table entries registered by the SDN controller. However, the observed delay is minimal. According to the QoS Class Identifier (QCI) classification for smart home services, the acceptable latency for real time communications is defined as 60 ms [14]. Therefore, the delay introduced by the proposed system remains within an acceptable range.

However, signature-based IDS like Snort has limitations. Specifically, signature-based IDS detects based on known attack patterns (signatures), making them unable to handle unknown attacks or zero-day attacks. Even when new types of malware or variants emerge, they cannot be detected unless their signatures are registered. Furthermore, if signature database updates are delayed or not properly managed, detection accuracy can decline. To address these challenges, minimizing damage through Defense in Depth is crucial.

6. Conclusion and Future Work

This study focuses on the potential for security risks to spread across the entire home network due to users' lack of awareness and knowledge about security threats. To address this issue, we propose a system that leverages SDN to apply MUD and integrates an intrusion detection system to monitor communications. Upon detecting an attack, the system isolates the compromised device to minimize damage. Through a comparative analysis of device-to-device communication, we demonstrate that the proposed system maintains security requirements within the home network while ensuring acceptable communication performance.

As a future prospect of this study, it is necessary to conduct further performance evaluations of the proposed system in environments that more closely resemble real world scenarios. This includes testing with an increased number of devices and assessing latency when retrieving MUD files from remote servers, because the processing load on the SDN controller and IDS may increase as the network expands. These evaluations will help validate the effectiveness and feasibility of the proposed system under practical conditions.

Acknowledgements

This work was partly supported by JSPS KAKENHI Grant Number JP24H00698.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Sinha, S. (2025) State of IoT 2025: Number of Connected IoT Devices Growing 14% to 21.1 Billion Globally. IOT ANALYTICS (on-line). https://iot-analytics.com/number-connected-iot-devices/

[2] Touqeer, H., Zaman, S., Amin, R., Hussain, M., Al-Turjman, F. and Bilal, M. (2021) Smart Home Security: Challenges, Issues and Solutions at Different IoT Layers. *The Journal of Supercomputing*, **77**, 14053-14089. https://doi.org/10.1007/s11227-021-03825-1

[3] Schiller, E., Aidoo, A., Fuhrer, J., Stahl, J., Ziörjen, M. and Stiller, B. (2022) Landscape of IoT Security. *Computer Science Review*, **44**, Article ID: 100467. https://doi.org/10.1016/j.cosrev.2022.100467

[4] Ferrara, P., Mandal, A.K., Cortesi, A. and Spoto, F. (2020) Static Analysis for Discovering IoT Vulnerabilities. *International Journal on Software Tools for Technology Transfer*, **23**, 71-88. https://doi.org/10.1007/s10009-020-00592-x

[5] Bessler, M., Sangster, P., Upadrashta, R. and O'Connor, T. (2024) Hardening the Internet of Things: Toward Designing Access Control for Resource Constrained IoT Devices. *Proceedings of the* 17*th Cyber Security Experimentation and Test Workshop*, Philadelphia, 13 August 2024, 1-7. https://doi.org/10.1145/3675741.3675744

[6] Zheng, S., Apthorpe, N., Chetty, M. and Feamster, N. (2018) User Perceptions of Smart Home IoT Privacy. *Proceedings of the ACM on Human-Computer Interaction*, **2**, 1-20. https://doi.org/10.1145/3274469

[7] Hammi, B., Zeadally, S., Khatoun, R. and Nebhen, J. (2022) Survey on Smart Homes: Vulnerabilities, Risks, and Countermeasures. *Computers & Security*, **117**, Article ID: 102677. https://doi.org/10.1016/j.cose.2022.102677

[8] Lear, E., Droms, R. and Romascanu, D. (2019) Manufacturer Usage Description Specification. Internet Engineering Task Force (IETF), RFC 8520.

[9] Souppaya, M., Montgomery, D., Polk, T., *et al.* (2021) Securing Small-Business and Home Internet of Things (IoT) Devices: Mitigating Network-Based Attacks Using Manufacturer Usage Description (MUD). Special Publication (NIST SP) 1800-15, National Institute of Standards and Technology.

[10] Ranganathan, M., Montgomery, D., Ilias, O. and Mimouni, E. (2019) Soft MUD: Implementing Manufacturer Usage Descriptions on OpenFlow SDN Switches. *International Conference on Networks* (ICN) 2019, Valencia, 25-29 March 2019, 49-54.

[11] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., *et al.* (2008) OpenFlow: Enabling Innovation in Campus Networks. *ACM SIGCOMM Computer Communication Review*, **38**, 69-74. https://doi.org/10.1145/1355734.1355746

[12] Waleed, A., Jamali, A.F. and Masood, A. (2022) Which Open-Source IDS? Snort, Suricata or Zeek. *Computer Networks*, **213**, Article ID: 109116. https://doi.org/10.1016/j.comnet.2022.109116

[13] Salman, O., Elhajj, I., Chehab, A. and Kayssi, A. (2017) Software Defined IoT Security Framework. 2017 *Fourth International Conference on Software Defined Systems* (*SDS*), Valencia, 8-11 May 2017, 75-80. https://doi.org/10.1109/sds.2017.7939144

[14] Jang, H.C., Huang, C.W. and Yeh, F.K. (2016) Design a Bandwidth Allocation Framework for SDN Based Smart Home. 2016 *IEEE* 7*th Annual Information Technology, Electronics and Mobile Communication Conference* (*IEMCON*), Vancouver, 13-15 October 2016, 1-6. https://doi.org/10.1109/iemcon.2016.7746320