A Communication System with Equipment's Characteristics

Chia-Chun Lai, I-Hsien Liu, Chi-Che Wu, Jung-Shian Li
Department of Electrical Engineering / Institute of Computer and Communication Engineering,
National Cheng Kung University
No.1, University Rd., East Dist., Tainan City 70101, Taiwan

Chuan-Gang Liu*
Department of Applied Informatics and Multimedia, Chia Nan University of Pharmacy & Science
No.60, Sec. 1, Erren Rd., Rende Dist., Tainan City 71710, Taiwan
*E-mail: {cclai, ihliu, ccwu1988}@cans.ee.ncku.edu.tw, jsli@mail.ncku.edu.tw, chgliu@mail.cnu.edu.tw**
www.cans.ee.ncku.edu.tw, www.cnu.edu.tw

Abstract

Over the past few years, applications on the internet have grown rapidly. In order to identify a specific device, we usually use the information which attains from network packets such as IP address, MAC address and communication port, etc. However, using this kind of information is not enough to identify precisely. Therefore, our research focuses on characteristics of communication devices that can identify precisely and also create a communication system which is able to imitate these characteristics.

Keywords: ICS, Communication, Network, Device Identification, Cyber Security

1. Introduction

In the last few decades, communication technology has grown rapidly and the internet is widely used in our daily lives. The progress result in devices that can connect to the internet have an explosive increase. Compare to the past, some information such as IP address, MAC address, and communication port is commonly used to identify a specific device on the internet. However, using this kind of information is not enough to identify comprehensively because the information can simply be imitated by any other devices. In our research, we focus on the characteristics of communication device. And we also provide a mechanism that can analyze these characteristics and generate configuration references which other devices can apply. With the customized communication module and the references mentioned above, we can make other devices imitate the behavior of the analyzed device and provide a better effect on device emulation.

2. Background

In this section, we are going to discuss some researches and methods which are commonly applied to perform device identification.

2.1. *Packet Header*

When devices communicate with each other, they must use the same protocol. Thus, allow them to understand what the other side transmitted. Nowadays, the most commonly used protocol is the TCP/IP protocol suite. This protocol suite includes a great deal of widely used protocols. Some well-known protocols in the TCP/IP suite are IP, TCP, UDP, etc., which are performed in different layers. These protocols have their own specific packet header when they are used.

Some information in these headers or the combination of this information in the different header can be used to identify which device sends this packet. In the next few paragraphs, we are going to explain the detail of how information in headers are used for device identification.

2.1.1. *IP header*

IP is the principal protocol in network communication. It can simply be divided into two versions: IPv4 and IPv6. Although the packet header of IPv4 and IPv6 is different, they still have some similarities. In the IP header, the major fields are source address and destination address, both of them exist in IPv4 and IPv6 header. The main function of these headers is to identify which device sends out this packet and which device should this packet send to. With these two pieces of information, we could simply identify a specific device, but it's not enough for precise identification. Some networks may use NAT (Network Address Translation) and many devices would share a public IP address. In this situation, using an IP address as the only identifier is not enough for device identification.

2.1.2. *TCP header*

TCP often uses on reliable transmissions such as a webpage, video stream, and file transfer. In the TCP header, there are two fields that can be used for device identification. These two fields are the source port and destination field, but we can't use them independently. We need to combine port information with IP information mentioned in the previous paragraph to carry out a better performance for device identification.

2.1.3. *Ethernet frame*

Ethernet protocol is a protocol that belongs to the data link layer. Recently, it is the principle protocol in our daily internet environment. Each Ethernet packet has an Ethernet frame that contains a header. An Ethernet header has four major fields: destination MAC addresses, source MAC addresses, Ethertype and IEEE 802.1Q tag or IEEE 802.1ad tag.

2.2. *Traffic Patterns*

Various devices will generate different traffic patterns depending on their specific needs. Some researches use traffic patterns to identify devices. Research from Hiroki KAWAI has shown that they can analyze traffic patterns and identify devices in different categories[1]. Also, there are some researches use machine learning to identify IoT devices which can classify all devices that connect to the network into some specific types[2,3,4].

3. Characteristics of Devices

Each device has its special characteristics when they are communicating. In our research, we focus on the behavior of devices when they respond or transmit packets. In our previous research, we obtain that the latency of the response packet between each device is unique. This discrepancy can cause by various factors in different layers.

3.1. *Physical Layer*

The devices produced by the same factory with different models will have their unique characteristics due to the variation of assembly lines. If we focus on the same model devices, they still exist unique characteristics because of the standard error of the machine which is allowed by its supplier. Even more, the manufacturing tolerance which is set by the manufacturer will also increase the diversity of characteristics in devices.

3.2. *Transport Layer*

When devices communicate on the internet they have to determine which protocol to be used in the transport layer. Using different protocols will result in respective behavior therefore generate a distinctive characteristic of devices. For instance, in a general WAN internet environment, if we use UDP protocol for communication, it might provide lower latency than using TCP protocol5. This situation shows that which protocol is used will have a significant impact on the characteristics of devices.

3.3. *Application Layer*

Every device has its own operating systems. Also, the executive application on the device would also affect the behavior. Even the same function programmed in a different language would lead to various characteristics of devices. Furthermore, regardless of having the same model device, if they are running in a different version of firmware, they may also increase the diversity of characteristics. In conclusion, the application layer has many factors that will affect the behavior of the device.

4. Experiment and Result

In our research, we propose a communication system that can analyze the characteristics of devices and then use these data to perform characteristics emulation of devices. We use a Schneider programmable logic controller (PLC) to be the device where we want to emulate and use a Windows 10 PC responsible for analyzing other devices. Owing to the device we used, the experiment will use a specific protocol which is commonly used in industrial control system called Modbus. Also, our communication system is programmed in C++ and run on Linux which can perform a better latency control of communication. Fig. 1. is the overview of our system. We can separate our system into two parts. In Fig. 2 and Fig. 3, we can see it have two major chains. The first chain is responsible for device analyzation and the second chain is responsible for emulation of the device.

4.1. *Analysis Side*

On the analysis side, we use a computer to send the response request to the device being analyzed. When the device response to our computer, we will record its round-trip time. Then we sorted out these data into a characteristic model and arrange a configuration file that can be used for the emulation side.

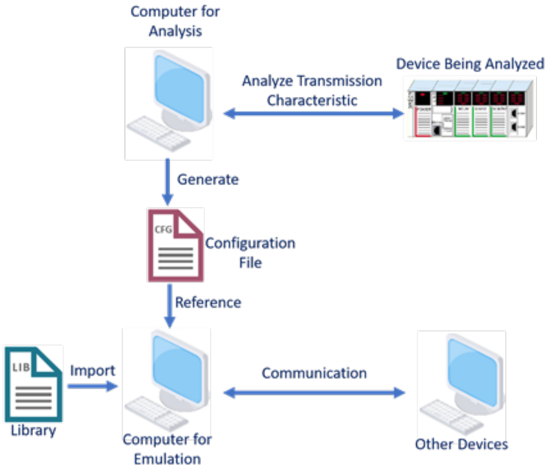

Fig. 1. Overview of our specific communication system

4.2. *Emulation Side*

On the emulation side, a device can either run our specific communication program on it or import our customize program library. Both of them can load the characteristic model generated by the analysis side. Furthermore, we add in a specific parameter called the loading parameter. Using the loading parameter, we can imitate the characteristics of the device. Meanwhile, the device is at working status.

4.3. *Analysis result*

In order to analyze the result of our emulation, our result focuses on the latency of response. In the result analysis, we use the Kolmogorov–Smirnov test for analyzation. Table 1 is the statistical table of our experiment, we let our analysis computer sending 10000 read register Modbus packet to both Schneider PLC and our emulator. In the Kolmogorov-Smirnov test, we assume that two samples come from the same distribution. Table 2 is our Kolmogorov-Smirnov test result. In the Kolmogorov-Smirnov test result, we can obtain a field that is asymptotic significance. In general, if asymptotic significance is greater than 0.05 or 0.1 then we accept the assumption of the Kolmogorov-Smirnov test, which means the two samples come from the same distribution. In table 2 we can see that our asymptotic significance is 0.997 which is greater than 0.01. As a result, we can say our emulation can respond to the same characteristics as the true Schneider PLC.

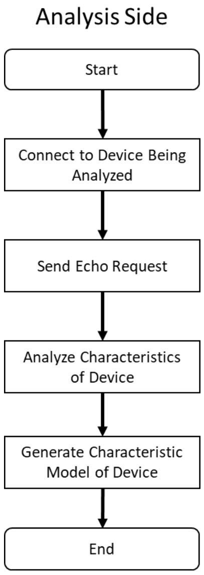
Analysis Side
Start
Connect to Device Being Analyzed
Send Echo Request
Analyze Characteristics of Device
Generate Characteristic Model of Device
End

Fig. 2. Flow chart of the analysis side of specific communication system

Table 1. The statistical table of experiment device

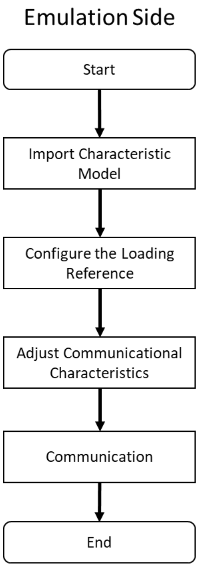
Emulation Side
Start
Import Characteristic Model
Configure the Loading Reference
Adjust Communicational Characteristics
Communication
End

Fig. 3. Flow chart of the emulation side our specific communication system

Device	Sample Size	Average	Standard Deviation
Schneider PLC	10000	5.9258	0.378033
Emulator	10000	5.9304	0.391371

Table 2. The Kolmogorov-Smirnov test result

Most Extreme Differences	Absolute	.006
	Positive	.001
	Negative	-.006
Kolmogorov-Smirnov Z		.403
Asymp. Sig. (2tailed)		.997

5. Conclusion

In this paper, we propose a mechanism to improve the emulation of device as well as focus on the difference of behavior on devices. Thus, we analyze the response latency of device, then create a characteristics model which can be used in our specific commutation system. Our specific communication system can load the model file and provide a load parameter that can imitate the loading of device and imitate the behavior of device which is at work. Also, using the statistical method we can show the result that using our system, we can let normal computer imitate the same behavior as the working PLC.

Acknowledgements

This work was supported by the Ministry of Science and Technology (MOST) in Taiwan under contract numbers MOST 109-2218-E-006-014-, MOST 109-2218-E-006-510- and MOST 109-2221-E-041-001-.

References

1. H. Kawai, S. Ata, N. Nakamura and I. Oka, Identification of communication devices from analysis of traffic patterns, *2017 13th International Conference on Network and Service Management (CNSM),* Tokyo, Japan, Nov. 2017, pp. 1-5
2. Ola Salman, Imad H. Elhajj, Ali Chehab, Ayman Kayssi, A machine learning based framework for IoT device identification and abnormal traffic detection, *Trans Emerging Tel Tech.,* 2019
3. Antônio J. Pinheiro, Jeandro de M. Bezerra, Caio A.P. Burgardt, Divanilson R. Campelo, *Identifying IoT devices and events based on packet length from encrypted traffic,* Computer Communications, vol. 144, 2019, pp. 8-17
4. Yair Meidan, Michael Bohadana, Asaf Shabtai, Juan David Guarnizo, Martín Ochoa, Nils Ole Tippenhauer, and Yuval Elovici, ProfilIoT: a machine learning approach for IoT device identification based on network traffic analysis, *SAC '17: Proceedings of the Symposium on Applied Computing,* New York, United States, Apr. 2017
5. I. Coonjah, P. C. Catherine and K. M. S. Soyjaudah, Experimental performance comparison between TCP vs UDP tunnel using OpenVPN, *2015 International Conference on Computing, Communication and Security (ICCCS)*, Pamplemousses, Mauritius, Dec. 2015, pp. 1-5