

# POSTER: A Framework for IoT Reprogramming

Nian Xue<sup>1,2</sup>, Lulu Liang<sup>3</sup>, Jie Zhang<sup>1,2</sup>, and Xin Huang<sup>1</sup>(✉)

<sup>1</sup> Department of Computer Science and Software Engineering,  
Xi'an Jiaotong-Liverpool University, Suzhou, China

[Nian.Xue15@student.xjtlu.edu.cn](mailto:Nian.Xue15@student.xjtlu.edu.cn), [Xin.Huang@xjtlu.edu.cn](mailto:Xin.Huang@xjtlu.edu.cn)

<sup>2</sup> School of Electrical Engineering and Electronics and Computer Science,  
University of Liverpool, Liverpool, UK

[Jie.Zhang3@liverpool.ac.uk](mailto:Jie.Zhang3@liverpool.ac.uk)

<sup>3</sup> China Information Technology Security Evaluation Center, Beijing, China  
[lianglulu@secemail.cn](mailto:lianglulu@secemail.cn)

**Abstract.** The OpenFlow protocol, as a fundamental element for Software Defined Networking (SDN) architecture, only supports for packet forwarding across switches in general networks. In this paper, the authors propose Software-Defined Function (SDF) which entitles administrators to manage the Internet of Things (IoT) devices and services through abstraction of the underlying infrastructure. The authors further present OpenFunction, a secure communications protocol stemmed from OpenFlow, which enables the IoT devices to be upgraded or reprogrammed remotely and securely. Finally, the authors implement a preliminary SDF system and evaluate its performance. The experimental results demonstrate that the SDF and OpenFunction can grant programmability, flexibility, centralization and security to the IoT.

**Keywords:** OpenFunction · Security · IoT · Software Defined Function

## 1 Introduction

The recent burgeoning of the Internet of Things (IoT) has been attracting an increasing number of researchers and experts with great attentions due to its significant economic and social values [1]. However, in the wake of the incremental number of the IoT devices, it is more and more difficult to manage and maintain so many devices efficiently [4,5]. In the meantime, as current IoT devices become considerably intelligent and heterogeneous [6], people thus expect more capabilities and features from these devices. According to [2], there is a tendency that the IoT nodes will receive software updates more frequently due to the growingly dynamic and changeful requirements and services from users and enterprises. Thereupon, how to reprogram or update the remote IoT devices securely and timely is another ongoing challenge.

In order to address the problems mentioned above, this paper proposes Software Defined Function (SDF) that utilizes an SDF controller in the control layer, via a function station situated in the physical infrastructure layer, to reprogram

the corresponding IoT end device through OpenFunction protocol derived from OpenFlow [3]. In particular, two specific security protocols of OpenFunction are designed to assure the secure communication between an SDF controller and a Function Station. Lastly, a demo system is realized and performance is evaluated in the archetypal SDF system.

## 2 System Design Overview

In our proposed framework, it comprises three kinds of components: *SDF Controller*, *Function Station* and *IoT Device*. The controller and the function station are usually connected by the Internet or LAN, while the IoT devices are often connected to the function station through short range wireless or wire manners such as WiFi, Bluetooth, Zigbee, serial line and so on. Below is the system design.

- **IoT device.** IoT devices in our framework are low-price and low-energy ones, mainly responsible for collecting or sending data. They consist of various smart entities, for example, temperature sensor, noise sensor, PM 2.5 sensor, etc.
- **Function station.** A function station connects to IoT devices and a controller simultaneously in the middle, responsible for upgrading or reprogramming the IoT devices according to the instruction from the controller.
- **SDF controller.** An SDF controller plays a pivotal role in the SDF framework, similar to the function of a human brain. Since it is able to remotely upgrade or reprogram the functions in IoT devices via a function station, it is unnecessary to deploy the controller near those IoT end devices.
- **Protocol I: OpenFunction Authenticated Handshake.** The primary aim is to provide authentication between a controller and function stations. In the first two steps, IDs and nonces are exchanged. Then the function station and the controller negotiate a session key using the pre-distributed public key with an SSL (Secure Socket Layer)-like procedure.
- **Protocol II: OpenFunction Messaging.** It is used to transmit the OpenFunction\_reprogramming messages. Messages are encrypted using the session key; and its authenticity is guaranteed with a message authentication code. After the above processes, the function station can reprogramme the IoT device according to the message content. Figure 1 below shows the whole process.

## 3 Evaluation and Implementation

A function station has stored beforehand a series of functions like a function warehouse. After receiving the update instructions from the authenticated controller, the function station will reprogramme the specified smart device through Avrdude, a program used to burn a hexadecimal coding into a firmware. In our experiment, one IoT device was originally preprogrammed as a temperature sensor, as shown in Fig. 3(a). After the experiment, its function has been

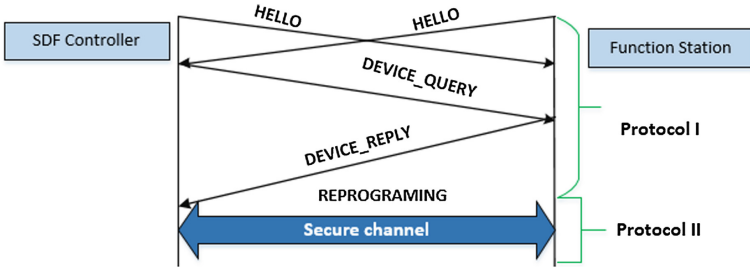


Fig. 1. Process of OpenFunction.

```

PMK: 89272774
b1f8e6048ea2c7287ae4f49f32053bb76fcd2d3b998215d0d9075a2aab14e1c5
success
OpenFunction_Reprogramming
key_1: 2693a337e3921b0d8f3a89aba15795866d8800361a74a8fce0db013ee9753350
key_2: 7fb7cb408db0413adb415c24fae3933ca29e2e206cb107ab6ba987804301ebdd
AESHEXCODE: e5fe7872c8f1cc63ff8ab89380f9d246
REPROGRAMMING_MAC
50a7db4cf68453f759fcf17011ba0aa1f3bf17d426cb9733ef9f055ede0bca7b
    
```

(a) Experiment result on controller

```

success
PMK: 89272774
b1f8e6048ea2c7287ae4f49f32053bb76fcd2d3b998215d0d9075a2aab14e1c5
OpenFunction_REPROGRAMMING
key_1: 2693a337e3921b0d8f3a89aba15795866d8800361a74a8fce0db013ee9753350
key_2: 7fb7cb408db0413adb415c24fae3933ca29e2e206cb107ab6ba987804301ebdd
REPROGRAMMING_MAC
50a7db4cf68453f759fcf17011ba0aa1f3bf17d426cb9733ef9f055ede0bca7b

avrdude: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.00s

avrdude: Device signature = 0x1e950f
avrdude: NOTE: "flash" memory has been specified, an erase cycle will be performed
To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "ss.hex"
avrdude: writing flash (5868 bytes):

writing | ##### | 100% 0.98s

avrdude: 5868 bytes of flash written
avrdude: verifying flash memory against ss.hex:
avrdude: load data flash data from input file ss.hex:
avrdude: input file ss.hex contains 5868 bytes
avrdude: reading on-chip flash data:

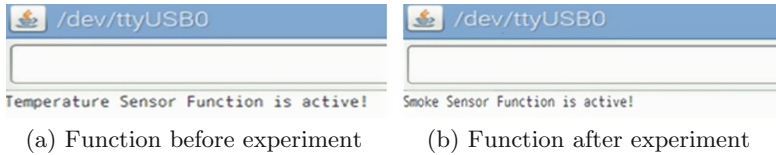
Reading | ##### | 100% 0.79s

avrdude: verifying ...
avrdude: 5868 bytes of flash verified

avrdude: safemode: Fuses OK (E:00, H:00, L:00)
avrdude done. Thank you.
    
```

(b) Experiment result on function station

Fig. 2. Implementing results of OpenFunction



**Fig. 3.** Reprogramming results of the IoT device (Arduino)

reprogrammed to a smoke sensor. Results of this experiment are illustrated in Figs. 2(a) and 3(b). From the figures we can find that the function has been changed. The average runtime for Protocol I is 35.01 ms, and for Protocol II is 31.33 ms. The overhead costs are acceptable.

## 4 Conclusion

In this paper, we have proposed and implemented SDF, a secure framework for reprogramming IoT devices. Two protocols are designed to guarantee the security during the reprogramming process. Test result indicates OpenFunction can be used to support IoT devices, as well as obtaining flexibility and security.

**Acknowledgments.** This work has been supported by the XJTU research development fund projects RDF140243, as well as by the Suzhou Science and Technology Development Plan under grant SYG201516, and Jiangsu Province National Science Foundation under grant BK20150376. This work has been supported in part by the Natural Science Foundation of China under Grant No. 61401517, in part by the National High Technology Research and Development Program (“863” Program) of China under Grant No. 2015AA016001.

## References

1. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT) a vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **29**(7), 1645–1660 (2013)
2. Huth, C., Duplys, P., GNeysu, T.: Secure software update and IP protection for untrusted devices in the Internet of Things via physically unclonable functions. In: 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops) (2016)
3. Mckeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J.: OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Comput. Commun. Rev.* **38**(2), 69–74 (2008)
4. Xu, R., Huang, X., Zhang, J., Lu, Y., Wu, G., Yan, Z.: Software defined intelligent building. *Int. J. Inf. Secur. Priv. (IJISP)* **9**(3), 84–99 (2015)
5. Xue, N., Huang, X., Zhang, J.: S<sup>2</sup>Net: a security framework for software defined intelligent building networks. In: *The IEEE International Conference on Trust, Security and Privacy in Computing and Communications* (2016)
6. Wang, D., Lo, D., Bhimani, J., Sugiura, K.: AnyControl - IoT based home appliances monitoring and controlling. In: *IEEE Computer Software and Applications Conference* (2015)