The Implementation of zk-SNARK HH Authentication on IoT Protocols

https://doi.org/10.3991/ijoe.v18i06.28893

Ari Kusyanti(✉), Nurudin Santoso, Puspita Ainunnazah, Luthfi Maulana
Department of Information Technology, Universitas Brawijaya, Malang, Indonesia
ari.kusyanti@ub.ac.id

Abstract—CoAP and MQTT are messaging protocols in the application layer of the IoT architecture. Both protocols, in general, have not been equipped with certain authentication schemes to cope with the network security gap. Without an authentication mechanism, the protocol is prone to mand-in-the middle attacks. Addressing this issue, the present study attempted to implement an authentication scheme using zk-SNARK Homomorphic Hiding (HH). This authentication mechanism does not require a token exchange and is safe from an eavesdropper. In this study, zk-SNARK HH was implemented to CoAP and MQTT protocols. The performance of both protocols was compared. The result of the study indicates that zk-SNARK was successfully implemented in CoAP and MQTT protocols. The result of the independent sample t-test presented a significant difference in COAP and MQTT memory usage where MQTT required larger memory. Regarding time execution, no significant difference was noticed. To conclude, CoAP may serve as an alternative when zk-SNARK is implemented in a device with limited memory as it requires lower memory and has an execution time that is not significantly different from MQTT.

Keywords—authentication, IoT, zk-SNARK, Homomorphic Hiding, MQTT, CoAP

1 Introduction

Internet of things (IoT) allows users to manage several devices remotely simultaneously through the internet [1] and applied in various applications such as in health [2], education [3], agriculture [4] and so on. All components are linked through communication in the form of data and information exchange between devices based on rules, format, and functions understood by the devices. This data exchange rule is called the messaging protocol.

Among several IoT messaging protocols are Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP) [5]. MQTT is an IoT protocol designed as a light data transmission protocol, offering advantages in IoT devices. Among its advantages is minimizing IoT devices' network bandwidth and resources while ensuring the accuracy and timeliness of the data transfer [6]. Meanwhile, CoAP

refers to a web-based data transfer protocol designed for nodes and networks with limited resources.

An IoT system requires a security scheme to protect its data from various possible network security gaps. One of the important factors of network security lies in the authentication. Authentication is applied to ensure the correctness of the sender's identity and the authenticity of the data being sent. Without an authentication mechanism, an attacker can pretend to be an authentic sender to see, modify, or even exploit the available data and causing troubles in the data exchange process, which may also lead to the system performance failure. Once an attacker gains access to control a system, they can take over the entire system.

Some previous studies have implemented authentication mechanisms on IoT protocols based on cryptography. The study conducted by [7] reports an implementation of lightweight crypto, while studies by [8] and [9] apply Elliptic Curve cryptography. Another study [10] used threshold cryptography. Meanwhile, the study conducted by [11] proposes a lightweight security scheme using the AES algorithm. The study reports that the proposed scheme (Auth-Lite) is ideal for securing CoAP systems using low power. Another study [12] developed an authentication mechanism that changes the password set for each authentication session. The developed mechanism is reported to manage securing IoT devices from attacks. A study by [13] designs and implements a token-based authentication in MQTT using four components: publisher, subscriber, broker, and token authentication server.

The study conducted by [14] applied OAuth, while [15] and [16] applied Digital Certificate for the authentication process in IoT. In [17] credential shared key is used, while Kerberos has also been implemented for IoT-based smart homes [18]. However, the existing literature mostly used tokens as authentication. This authentication method exchanges a secret token to be validated. This method still has a security gap in CVE-2019-13483. In this gap, attackers can read and duplicate the token through the sniffing process. This condition is worsened by the absence of a mechanism capable of validating the token sent to the verifier, allowing unexpected devices to easily use the token to pass the authentication and enter the system.

Regarding the problem described previously, an authentication method for an IoT system called Zero-Knowledge Proof (ZKP) is proposed. ZKP possesses a concept to prove a secret without requiring token exchange between Prover and Verifier. ZKP can be implemented using various rules or protocols, including Zero-Knowledge Zero-Knowledge Succinct Non-Interactive Argument of Knowledge Homomorphic Hiding, known as zk-SNARK HH, an authentication mechanism that does not exchange any sensitive data between two parties. zk-SNARK HH offers a more secured authentication mechanism as it does not involve confidential data exchange [19]. Based on the previous studies described earlier, the present study attempted to implement a security authentication using the zk-SNARK HH algorithm on an IoT system. The system employed CoAP and MQTT as its data exchange protocol, run upon the 6LoWPAN network. This system contains three nodes where two of them serve as sensor gateways while another node serves as a middleware. CoAP protocol was implemented in one of the sensor gateways, while MQTT protocol was implemented to the other sensor gateway to send the data to the middleware.

2 Zero-knowledge succinct non-interactive argument of knowledge Homomorphic Hiding

Zero-knowledge Proof (ZKP) is an authentication security method using polynomial probability theory. In ZKP system, a party is obliged to follow the protocol and rules determined by another party to see the correctness of a sensitive data without directly discover the secret. In other words, ZKP is a cryptographic method used to prove a secret without involving the process of exchanging the secret. In the present study, zk-SNARK HH was applied as the ZKP method [19].

The core element of this method lies in the use of Homomorphic Hiding (HH) that allows mathematical operation of an encrypted values. For instance, the x and y values are known, and the values are encrypted, thus:

$E(x)$ and $E(y)$

Homomorphic Hiding zk-SNARK should meet the following properties:

- For most x, it is difficult to find x if the $E(x)$ is known.
- Different input results in different output – thus if $x \neq y$, then $E(x) \neq E(y)$.
- If the $E(x)$ and $E(y)$ are known, the $E(x + y)$ values can be determined from E(x) and $E(y)$.

Homomorphic Hiding zk-SNARK is applicable for Zero-Knowledge proof. For instance, Prover attempts to prove to the verifier that they know the number x, y, thus x+y with the prime value p and generator g.

1. The prover sends $E(x)$ and $E(y)$ to the Verifier:

$$E(x) = g^x \ mod\ p \text{ and } E(y) = g^y mod\ p \tag{1}$$

2. Verifier calculates $E(x + y)$ that can be done because $E(x)$ is HH with discrete logarithm:

$$E(x + y) = g^{x+y}\ mod\ p - 1$$

$$E(x + y) = g^x g^y \tag{2}$$

3. The verifier also calculates E(x+y) and checking whether:

$$E(x + y) = E(x).E(y) \tag{3}$$

Different encrypted inputs create ciphertext and verifier receives the proof only if th prover send ciphertext from x and y that is same with the $x + y$ encryption result. Verifier does not know x and y because the verifier only has an access to ciphertext [19]. In this study, the zk-SNARK HH algorithm is implemented on an IoT system. This system contains three nodes where two of them serve as sensor gateways while another node serves as a middleware. Sensor gateway has to authenticated itself to middleware before sending the sensor data, hence the middleware is a verifier that authenticate the sensor gateway that proves that they are legitimate sensor gateway.

3 System design

In this study, an authentication method for an IoT system called Zero-Knowledge Proof (ZKP) is implemented. ZKP possesses a concept to prove a secret without requiring token exchange between Prover and Verifier. The system consists of three nodes connected to the MRF24J4M0MA as the radio module functioning to connect each node. Two nodes serve as the sensor gateway while the other node serves as middleware. Sensor gateway acts as a prover, and the middleware as a verifier.

The sensor gateway in this study consists of two types, MQTT and CoAP protocols. The middleware serves as the service provider to receive data from both protocols applied in the system. In addition to performing the monitoring function, middleware in this study also serves as the broker for the MQTT protocol. The authentication process begins when the node (in this study, sensor gateway node), acting as the prover, asks for authentication access to the verifier (middleware). The verifier then gives the first challenge to the prover. The prover accepts the challenge and sends the answer to the verifier. Once the verifier receives the response, the verifier checks the correctness of the answer. The verifier grants connection access to provers if they manage to obtain a threshold value of 100% as depicted in Figure 1.

In this design, zk-SNARK HH is implemented on the prover and verifier. There are two values the prover and verifier should agree on, namely the divisor/modulus p of the random prime number p and randomly generated generator g. The authentication process begins when the sensor gateway node asks for access to authenticate the verifier (i.e., middleware). As a response, the verifier then gives the first challenge to the sensor node. The sensor node receives the challenge and responds in the form of ciphertext of encrypted x and y values based on equation (1), where the sum of x and y values is equal to the challenge given by the verifier. The ciphertext is sent back to the verifier. After receiving a response from the sensor node, the verifier calculates the ciphertext based on equation (2). This can be done due to the implementation of Homomorphic Hiding for zk-SNARK. The verifier should also calculate the ciphertext of the challenge it gives to provers. The sensor node is deemed successful in solving the challenge if the calculation result aligns with equation (3).

The sensor node gains a 50% trust level when it manages to solve the challenge correctly and a 0% trust level when it fails to solve the challenge. This process is performed continuously until the verifier's trust level in the sensor node reaches the determined threshold. In this study, the prover is required to solve challenges for 34 rounds to gain a 100% trust level. When the determined trust value is met, the verifier will allow the sensor node to send the data.

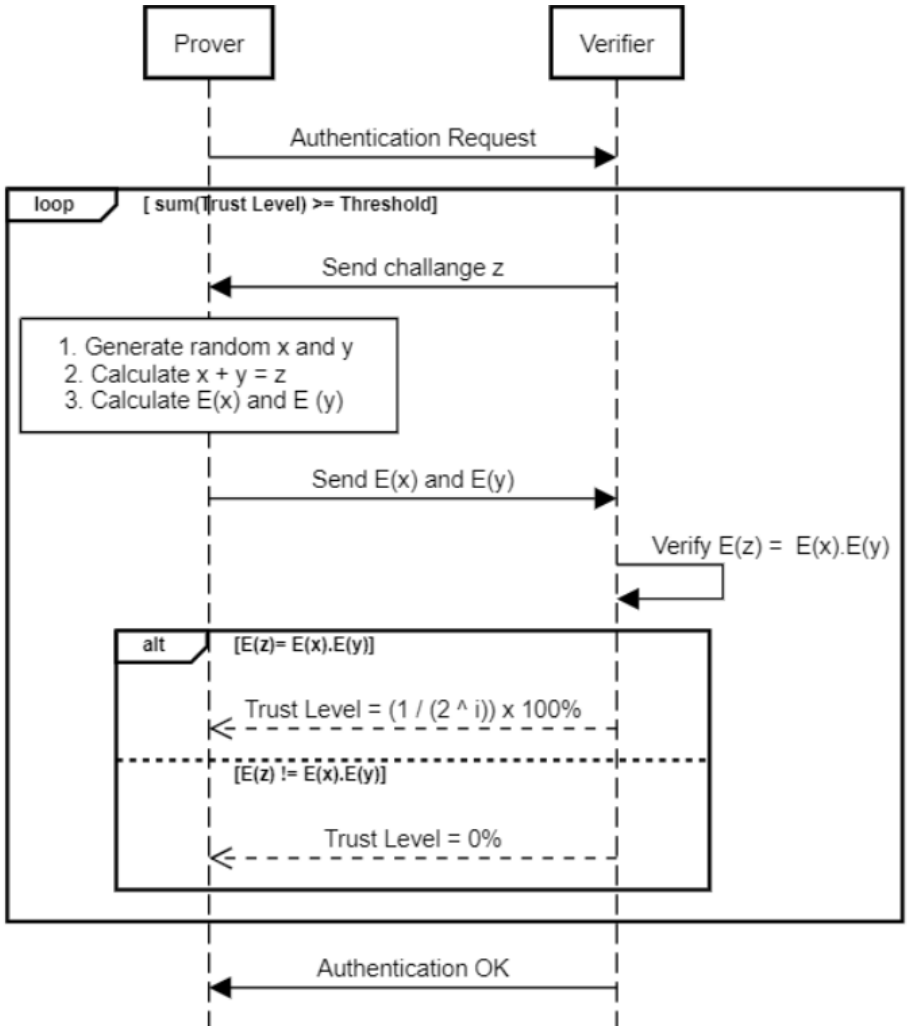

Fig. 1. Sequence diagram on zk-SNARK authentication process

4 Result and comparison

A performance test was conducted to find out the system performance when carrying out the authentication process. The performance test was conducted by considering IoT devices' limited resources. The test was performed following the test metric, i.e., dynamic memory usage and authentication process execution.

The memory usage and authentication time were used as parameters to find out the number of memories used during the authentication process and the length of the pro-

cess. Memory usage used as the metric in this test was the memory usage on the device's SRAM, while the execution time was counted from the time verifier receives a connection request from the prover until the trust level threshold was attained.

The following Tables 1 and 2 display the result of the performance test. The average memory usage on MQTT and CoAP was 0.485 MB and 0.283 MB, respectively. Regarding the authentication time, MQTT takes 0.317 ms on average, while CoAP took 0.216 ms on average, as displayed in Table 2.

Table 1. zk-SNARK authentication memory usage test

Parameter	Group	N	Mean	SD	SE
Memory	CoAP	51	0.283	0.096	0.014
	MQTT	51	0.485	0.091	0.013

Table 2. zk-SNARK authentication time test

Parameter	Group	N	Mean	SD	SE
Time	CoAP	51	0.261	0.543	0.076
	MQTT	51	0.317	0.359	0.050

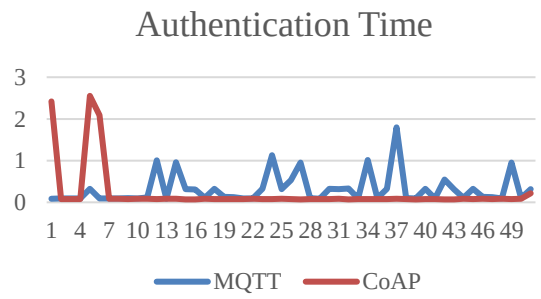

Fig. 2. Comparison of authentication time between CoAP and MQTT

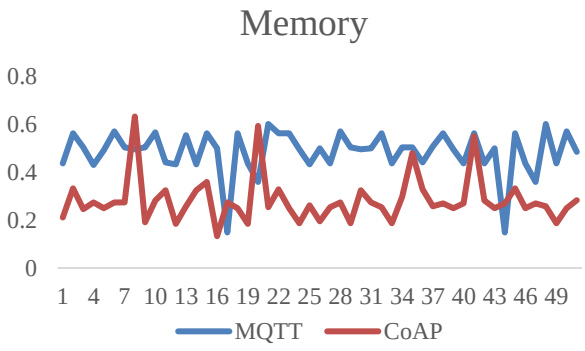

Fig. 3. Comparison of memory usage between CoAP and MQTT

The performance test results were statistically processed to interpret the data. The statistical process aim to find out the difference in zk-SNARK HH authentication on MQTT and CoAP protocols in terms of memory usage and execution time. An independent sample t-test was applied. Two hypotheses were proposed, H0 was accepted if there is no significant difference in memory usage and authentication time between the zk-SNARK HH authentication method on MQTT and CoAP. H0 was rejected if there is a significant difference in memory usage and authentication time between MQTT and CoAP protocols. The data were processed using JASP [20]. The results were presented in Tables 1 and 2.

The independent sample t-test result indicates a significant difference in memory usage between MQTT and CoAP protocols, shown by the Sig. value (2-tailed) of <0.001. However, the independent sample t-test found an insignificant difference in execution time between zk-SNARK authentication on MQTT and CoAP protocols, as indicated by the Sig. value (2-Tailed) of 0.268 (>0.05).

Table 3. Independent sample T-Test on execution time

Parameter	T	df	P
Time	-1.114	100	0.268

Table 4. Independent sample T-Test on memory usage

Parameter	T	df	P
Memory	-10.914	100	< 0.001

According to the independent sample t-test, there is a significant difference in memory usage when implementing zk-SNARK HH authentication on MQTT and CoAP protocols. It was found that MQTT requires more memory than CoAP. In other words, the CoAP protocol is deemed more effective when zk-SNARK authentication is implemented in a device with limited resources. The MQTT protocol requires more memory because the resource (source code and library on paho-mqtt) used in each device is more than the CoAP protocol (CoaPthon).

It was found that the MQTT protocol exhibited better performance in terms of time execution, thanks to its data exchange architecture (publish-subscribe). The subscriber nodes in MQTT do not require repeated requests to gain the data, allowing the subscribed node to automatically receive data from publishers according to the topic. In contrast, CoAP data exchange architecture (client-server) requires continuous data requests. Although the MQTT protocol is faster, the data concludes that no significant difference is found in terms of execution time. Thus, zk-SNARK HH can be implemented both on MQTT and CoAP protocol without a significant difference in execution time.

5 Conclusion

Based on the performance test, zk-SNARK HH authentication on CoAP and MQTT protocols was proven effective in IoT systems. There is a significant difference in COAP and MQTT memory usage, where MQTT requires larger memory. Regarding time execution, no significant difference was noticed. Thus, CoAP may serve as an alternative when zk-SNARK is implemented in a device with limited memory as it requires lower memory and has an execution time that is not significantly different from MQTT.

6 References

[1] Guo, B., Zhang, D., Yu, Z., Liang, Y., Wang, Z. and Zhou, X. (2012). From the internet of things to embedded intelligence. Xian, Republic of China: Springer, pp.399-420. https://doi.org/10.1007/s11280-012-0188-y

[2] Al-Mutairi, A. W., Al-Aubidy, K. M., & Al-Halaiqa, F. N. (2021). IoT-Based Real-Time Monitoring System for Epidemic Diseases Patients: Design and Evaluation. International Journal of Online and Biomedical Engineering (iJOE), 17(01), pp. 63–82. https://doi.org/10.3991/ijoe.v17i01.18849

[3] Moulay Taj, A., Chacon Sombria, J., Gaga, A., Abouhilal, A., & Malaoui, A. (2021). Conception and Implementation of an IoT System for Remote Practical Works in Open Access University's Electronic Laboratories. International Journal of Online and Biomedical Engineering (iJOE), 17(02), pp. 19–36. https://doi.org/10.3991/ijoe.v17i02.19755

[4] Hamdi, M., Rehman, A., Alghamdi, A., Nizamani, M. A., Missen, M. M. S., & Memon, M. A. (2021). Internet of Things (IoT) Based Water Irrigation System. International Journal of Online and Biomedical Engineering (iJOE), 17(05), pp. 69–80. https://doi.org/10.3991/ijoe.v17i05.22081

[5] Al-Masri, E., Kalyanam, K., Batts, J., Kim, J., Singh, S., Vo, T. and Yan, C., (2020). Investigating Messaging Protocols for the Internet of Things (IoT). https://doi.org/10.1109/ACCESS.2020.2993363

[6] Liu, X., Zhang, T., Hu, N., Zhang, P. and Zhang, Y. (2020). The method of Internet of Things access and network communication based on MQTT. Computer Communications, 153, pp.169-176. https://doi.org/10.1016/j.comcom.2020.01.044

[7] J. Lee, W. Lin, Y. Huang. (2014). A Lightweight Authentication Protocol for Internet of Things., International Symposium on Next-Generation Electronics, ISNE 2014. https://doi.org/10.1109/ISNE.2014.6839375

[8] G. Zhao, X. Si, J. Wang, X. Long, T.Hu, (2011). A Novel Mutual Authentication Scheme for Internet of Things., Proceedings of 2011. International Conference on Modelling, Identification and Control, Shanghai, China, June 26-29, 2011.

[9] F. Santoso, N. Vun, (2015). Securing IoT for Smart Home System. International Symposium on Consumer Electronics (ISCE) Securing, 2015 IEEE. https://doi.org/10.1109/ISCE.2015.7177843

[10] P. Mahalle, N. Prasad, R. Prasad. (2014). Threshold cryptography-based group authentication (TCGA) scheme for the Internet of things. https://doi.org/10.1109/VITAE.2014.6934425

[11] Ukil, Arijit & Bandyopadhyay, Soma & Bhattacharyya, Abhijan & Pal, Arpan & Bose, Tulika. (2014). Lightweight security scheme for IoT applications using CoAP. International

Journal of Pervasive Computing and Communications. 10. 372-392. https://doi.org/10.1108/IJPCC-01-2014-0002

[12] T. Shah and S. Venkatesan. (2018). Authentication of IoT Device and IoT Server Using Secure Vaults, 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), 2018, pp. 819-824. https://doi.org/10.1109/TrustCom/BigDataSE.2018.00117

[13] A. Bhawiyuga, M. Data and A. Warda (2017). Architectural design of token based authentication of MQTT protocol in constrained IoT device, 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA), 2017, pp. 1-4. https://doi.org/10.1109/TSSA.2017.8272933

[14] D. Rivera, L. Paris, G. Civera, E. Hoz, I. Maestre (2015). Applying an unified access control for IoT based Intelligent agent system.IEEE international conference on service-oriented computing and application. https://doi.org/10.1109/SOCA.2015.40

[15] M. Panwar, A. Kumar, (2015). Security for IoT an effective DTLS with public certificates. International conference on advances in Computer Engineering and application (ICACEA, 2015). https://doi.org/10.1109/ICACEA.2015.7164688

[16] A. Park, H. Kim. (2015). A framework of device authentication management in IoT environments*., 5th International Conference on IT Convergence and Security, ICITCS 2015 – Proceedings. https://doi.org/10.1109/ICITCS.2015.7292918

[17] P. Periera, J. Eliasson, J. Delsing. (2014). An authentication and access control framework for CoAP based internet of things. Proceedings, IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society. https://doi.org/10.1109/IECON.2014.7049308

[18] P. Gaikwad, J. Gabhane, S. Golait, (2015). 3-level secure Kerberos authentication for smart home system using IoT. International conference on next generation computing technologies 2015 (NGCT2015). https://doi.org/10.1109/NGCT.2015.7375123

[19] Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, Madars Virza. (2014). Succinct non-interactive zero knowledge for a von Neumann architecture. Proceedings of the 23rd USENIX conference on Security Symposium August 2014 Pages 781–796.

[20] Goss-Sampson, M. (2018). Statistical Analysis in JASP: A gudie for Studets. https://jasp-stats.org/

7 Authors

Ari Kusyanti is currently an Assistant Professor at the Information Technology Department of Universitas Brawijaya, Malang, Indonesia. Her research interest is computer security, data privacy and IoT security.

Nurudin Santoso is currently an Associate Professor at the Information Technology Department of Universitas Brawijaya, Malang, Indonesia. His research interest is software engineering.

Puspita Ainunnazah is graduated from Information Technology Department of Universitas Brawijaya, Malang, Indonesia. Her research interest is IoT security.

Luthfi Maulana is graduated from Information Technology Department of Universitas Brawijaya, Malang, Indonesia. His research interest is IoT security.

Article submitted 2021-12-15. Resubmitted 2022-01-29. Final acceptance 2022-03-03. Final version published as submitted by the authors.