ISSN: 2321-7782 (Online)
Impact Factor: 6.047

Volume 4, Issue 5, May 2016

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Securing Communication in Class-0 IOT Devices

Kavita Mittal
M. Tech Scholar, Department of Comp.Sci.a nd Engg.
Tula's Institute, the Engg. and Management College
Dehradun – India

Abstract: Class-0 devices are devices with limited resources such as CPU, memory (ROM and RAM), and battery life. These devices often function as sensors collecting information, machine to machine (M2M) or smart devices controlling electrical appliances and services. When these devices are connecting to a network they become known as "things" and termed as "Internet of Things" (IoT). IoT devices are connected to the internet to allow for the collection and exchange of data with web servers and cloud data centres. security is defined as the protection of data from unauthorized interference or monitoring by ensuring confidentiality, authenticity and integrity of data.

Keywords: Internet of Things, AES, Encryption, class-0 devices, HTTP, CoAP, TLS.

I. INTRODUCTION

The IoT is a new scientific assumption in IT arena. The two words, the first word is Internet and the second word is Things are joined together and known as Internet of Things which is also shortly well-known as IOT. The Internet of Things can also be observed as a global network that allow the communication between human-to-human, human-to-things and things-to-things, it might be anything in the world that provides unique identity for each and every object. Different IoT devices connected directly to an Internet router or by using an IoT gateway which acts as a bridge between the constrained IoT network and the internet.

A. Devices of Internet of Things

IoT is a network of objects such as embedded computers, controllable and intelligent automated devices (smart devices), and sensors, with the ability to connect and exchange data with other devices and services [1] [2]. IoT has many applications such as home automation, manufacturing, environmental monitoring, medical and health care systems, and transportation. Table 1 gives the classifications scheme to differentiate among IoT devices based on their system resources.

Table 1- Classes of Constrained Devices (Kb = 1024 Bytes)

Name	RAM	ROM
Class 0, (C0)	<<10Kb	<<100Kb
Class 1, (C1)	~ 10 Kb	~ 100 Kb
Class 2, (C2)	~ 50 Kb	~ 250 Kb

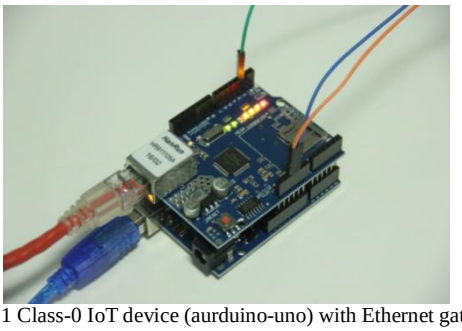

Figure 1 Class-0 IoT device (aurduino-uno) with Ethernet gateway

B. Protocols of IoT

- User Datagram Protocol (UDP). This protocol has significant benefits as it guarantees delivery of packages as well as order of delivery. At the same time, it requires a communication overhead for connection, additional resources to maintain the connection state and package delivery confirmation, which may cause timeouts due to the IEEE 802.15.4 network latencies.
- Datagram Transport Layer Security (DTLS). The main idea of the DTLS protocol is to use Transport Layer Security (TLS) over an unreliable datagram transport layer. DTLS features the same 4 sub protocols as TLS [4].
- Constrained Application Protocol (CoAP). On the application layer HTTP is commonly used for most applications that implement a client/server model. At the same time this protocol is supposed to run over reliable transport (like TCP) and cannot be used over UDP. Moreover, running HTTP protocol may require too much computational resources (like parsing HTTP headers, form parameters end, etc.) for a constrained device and does not take into account models that are used in the IoT (e.g. multicast, and unconfirmed requests). In order to overcome these issues a new protocol CoAP [3] is being designed generally as a subset of HTTP protocol that can be used over UDP transport. CoAP takes into account constrained computational resources of microcontrollers and scenarios of machine-to machine communications [3]. The main conceptual difference of CoAP from HTTP is a message abstraction that determines the type of request or response.

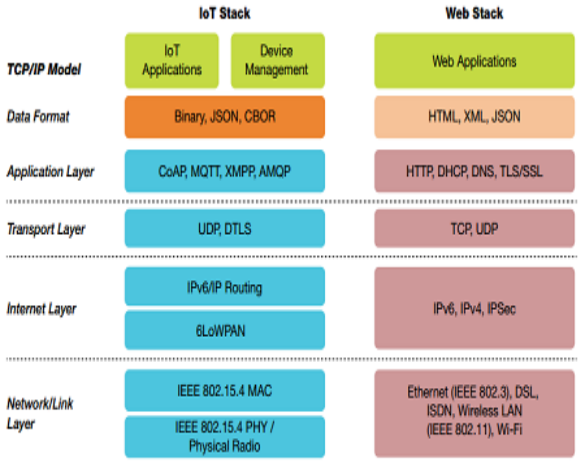

Figure 2 Comparison between web and IoT protocol stack

C. Security Challenges of IOT

As we increasingly connecting more and more devices with the Internet, new opportunities grow constantly to exploit the potential security vulnerabilities. Some IOT devices that are poorly secured could act as an entry points for cyber-attack so as allowing malicious individuals to make a device to malfunction re-program a device. Devices that are poorly designed can expose user data to theft by leaving data streams inadequately protected. Security vulnerabilities are also caused due to failing or malfunctioning of devices. These problems are much larger for the small, cheap, and ubiquitous smart devices in the Internet of Things.

II. SECURITY IN IOT

Security is defined as the protection of data from unauthorized interference or monitoring by ensuring confidentiality, integrity, and authenticity of data. Confidentiality of data is defined as the protection of data from disclosure to unauthorized persons, parties or systems. Integrity is defined as the preventions of falsification or modification of data by unauthorized persons. Authenticity refers to the verification of the identity of a device or system. [5] [6]

HTTP and HTTPS were not design for IoT devices with resource limitation and so a more efficient protocol was developed specifically for constrained resource devices. The Constraint Application Protocol (CoAP) is a specialized application layer protocol design for resource constrained devices such as IoT. In addition to provide security at different layers we need to provide security during communication. While communication is taking place in IoT enabled environment data should not hack and changed in between. In order to provides security during communication we use AES method to encrypt the data.

A. Adavanced Encryption Standard

AES is an Encryption standard that is based on a design principle termed as a substitution-permutation network, combination of both substitution and permutation, makes it fast in both software and hardware. AES does not use a Feistel network. AES is a variant of Rijndael which has a key size of 128 bits and fixed block size of 128 bits and 192 or 256 bits. By contrast, the Rijndael specification per set is specified with block and key sizes that may be any multiple of 32 bits, both with a minimum of 128 and a maximum of 256 bits.

B. Implementation of Cryptography in IoT

Transport Layer Security (TLS) works by using cryptography to ensure a secure a reliable connection for data communication. [23] Data is encrypted by the sender using the cryptographic public key of the recipient. The data is then sent across the internet to the recipient. Only the recipient's private key will be able to decrypt the data and this key is kept private and secure by the recipient. TLS is also used to provide a secure session by using asymmetric cryptography to secure exchange symmetric keys which are then used for the bulk of the data exchange. Asymmetric cryptography methods require more resources to operate than symmetric cryptography as security handshake and key exchange must take place. As Class-0 devices have limited resources asymmetric cryptography and (D)TLS protocols are too resource demanding for these devices [7].

Advanced Encryption Standard (AES) [7] is one such symmetric standard which operates at fast speeds and requires fewer resources than (D)TLS making it very suitable for Class-0 constrained devices [24]. AES inputs data as 16 bytes (128-bit) blocks which are then encrypted using a cryptographic key of 128-bit, 192-bit, and 256-bit in size [26]. The larger the key size the greater the security and resource requirement on the device to encrypt and decrypt.

III. RESEARCH METHODOLOGY

The proposed solution will take the form of a method involving symmetric and asymmetric encryption and a security gateway which serves as an intermediary among devices and the internet. Security processes which are too resource intensive for Class-0 devices are delegated to the gateway where data is processed into a secure form before it is being transmitted across the internet. This solution will provide confidentiality, integrity, and authenticity of data being transported across the internet and confidentiality of data as it passes between device and gateway. The solution will be designed to meet the requirements outlined in the objectives

Design Science Research Methodology Stages

- STEP 1: Identify Problem & Motivate-The first step in the design-science research methodology will be to define the problems in detail. The problems identified are centered on the security of data communications between device and

internet based destination. Many other security concerns may exist such as the security of data at rest but these are not discussed here as they are outside the scope of this research.

- STEP 2: Define Objectives-The objectives of the solution will be based on the identified problem from step 1. The objectives are the requirements which need to be met in order to solve the identified problems in Step 1.
- STEP 3: Design-In this stage requirements will be analyzed and development decisions will be made and written into the artifact design. This will be the blueprint on which the development process for implementing. Once the design is complete and mapped out a proof-of-concept (POC) will then be implemented from the finished design and prepared for demonstration and evaluation. This is an iterative process and with each stage a review of the requirements and objectives will be made to ensure the system development is on track. "A design artefact is complete and effective when it satisfies the requirements and constraints of the problem it was meant to solve."
- STEP 4-Implementation and Evaluation-Tests will be conducted on performance and effectiveness of the POC and data will be collected. The performance of the POC will be measured against how well it meets the objectives outlined in step 2 of the DSR process. Data will be collected on the resource requirements and processing times of the POC in the IoT device. Packet analysis will also be conducted to ensure data is secure at each stage of transfer. Conclusions will be drawn based on how effective the system performs against these criteria, and the pros and cons of the mechanism, standards and technologies used. It is hoped that lessons learned through this work will help others developing similar systems. The POC will then be evaluated to see how well it solves the problems identified in Step 1 and satisfies the design requirements outlined Step 3.
- STEP 5: Communication-The results from the research will be presented in the paper, then discussed and reflected upon. A presentation will also be conducted with supervisors and fellow students. It is hoped that these results and reflections will acts as helpful recommendations to those who read this paper. It is expected that some area requiring further research will be identified in this stage leaving the door open for other researchers to continue on from this work.

IV. Design of Security Solution

This part will concentrate on the plan of the security result. Those plan may be partitioned under three components: Device, IoT security Gateway, and Web server.

SECURING DEVICE TO GATEWAY

With the end goal a gadget on make arranged as Class-0 its asset must make underneath the asset edge as delineated Toward Bormann et al. [8] with short of what 100kb rom or less 10Kb ram. A gadget in those Arduino Uno (16MHz CPU, 32Kb RAM, 2Kb ROM) might make tasked on controls appliances, actuators, services, alternately gather information starting with sensors. Security from between gadget Also passage could a chance to be given utilizing equipment built symmetric encryption of the information join layer as and only those remote protocol (e.g. IEEE 802. 15. 4, IEEE 802. 11n). Remote security modes offer equipment AES symmetric encryption toward those information join layer. It depicts those encryptions Furthermore unscrambling from claiming information Similarly as it passes remotely from gadget should passage. Those systems for exchange relies on the accessible equipment. Remote transmission could a chance to be Gave utilizing an IEEE 802. 15. 4 modules for example, such that a ZigBee alternately 6LoWPAN interface at elective would accessible.

At interfacing with a framework units require help secured for a pre-shared enchantment (PSK) which will be presented around each appointed contraption in addition may be required to correspondence for the individual's section. Whatever unit units tuning for secured nearby with admiration to development won't need the ability ought further bolstering unscramble data without the PSK. Same duration of the time this protects data beginning with substances without the individuals PSK it abandons majority

of the data introduced accepting that an attacker manages with profession off those remote securities alternately make the individuals PSK beginning for an extra contraption.

SECURING DATA IN THE GATEWAY

Security conventions which need aid excessively overwhelming should run specifically starting with the Class-0 gadget are delegated of the passage. Those passage demonstrations Similarly as a go-between with plentiful assets to backing these efforts to establish safety Furthermore secure information in the recent past sending it through the web. Information sent from the IoT gadget will a chance to be sent of the passage utilizing conventions for example, CoAP Also Http What's more sent crosswise over the web utilizing HTTPS (HTTP over TLS) of the web server. Those payload of the packets will a chance to be formatted Concerning illustration JSON Questions What's more encrypted utilizing AES 128-bit alternately 256-bit symmetric encryption. This information item will exist inside the transmission payload same time those bundle header data for example, sourball Also end address stays unencrypted.

SECURING DATA FROM GATEWAY TO SERVER

Gateways are computational devices with enough resources to run operating systems and protocols necessary to securely transfer traffic across the internet. A gateway may take the form of a microcomputer with a Linux based operating system. An example of a microcomputer would be the RPi model B has a 700 MHz single core CPU, 512MB SDRAM, Ethernet port, and an SD-card reader as for on-board storage. The gateway has sufficient resources to apply heavier security and communication protocols which cannot be supported by the IoT device. An addition wireless transmitter can be attached to the RPi to connect wirelessly with IoT devices.

Data objects (sensor readings) are formatted as JSON and encrypted using the AES 128-bit before being transmitted to the gateway wirelessly. Wireless transmissions are secured using WPA2 PSK and AES 128bit PSK. Only authorized devices and the gateway possess the PSK so traffic is protected from attackers eavesdropping on wireless transmissions.

Once data is received by the gateway it is processed into HTTPS and prepared for transmission to the server. The gateway is configured with Secure Socket Layer (SSL) tools which are used to create a secure HTTPS connection between gateway and server. From this we can forward secure communications to the server over the internet using asymmetric cryptographic. Messages being transmitted to the server are encrypted with the server's public key which is installed in the gateway. Only the server can decrypt messages using its corresponding private key. The private key is located on the server and is not shared with any other devices. This maintains confidentiality of information as it passes over the internet. Once the HTTPS packets are received by the server they are decrypted using the private key. The encrypted data object can then be decrypted using the symmetric secret key from the originating device, in this case our class-0 IoT device. If the key is only present of a single IoT device and the server, it can be used to authenticate data received from either party. If the key is shared with multiple devices the devices are authenticated as part of a group. Data integrity can be provided at the object layer by providing a cryptographic hash of the data and encrypting it with the data before it is transmitted. The destination will be able to perform a check on data received from the IoT device and verify it matches the included hash. While this may be possible through the use of hashing algorithms such as SHA and MD5 it was not implemented in this solution and is an area for further work.

PROCESSING DATA SERVER SIDE

On the server side a RESTful web service is setup to receive information from the gateway. A secure HTTPS connection must be established between server and gateway. Data is received data through a secure socket (HTTPS) which is established using the server's privacy certificate. Once data is received the data contents are then extracted. The data is still encrypted using the device/server symmetric encryption key. Using the key and an AES algorithm the data is decrypted, processed and stored in a database. In order for a HTTPS connection to be established the server must first have a security certificate installed. Certificates

must also be signed by a trusted certificate authority in order to be trusted. Before the connection between the server and gateway security certificates received must be verified by a trust third party to ensure the connection is not compromised.

V. CONCLUSION

This research has focused on the security of data being communicated between highly constrained IoT devices and the internet. A Class-0 device does not have sufficient resources to support the transport layer security mechanism needed to securely transport data directly to across the internet. This paper attempts to build on existing research by addressing gaps in existing solutions. Using data object encryption and a An extra layer of protection is applied to data by encrypting its using AES before it leaves the IoT device. Using symmetric encryption confidentiality of data can be secured between device and the intended server destination. Sensitive data being passed to the network.

References

1. N Nurseitov, M Paulson, R Reynolds, and C Izurieta, "Comparison of JSON and XML Data Interchange Formats: A Case Study," in CAINE, 2009, pp. 157-162.
2. M. Vucinic, Grenoble Alps Univ., Grenoble, France Grenoble Inf. Lab., B. Tourancheau, F. 56 Rousseau, and A. Duda, "OSCAR: Object security architecture for the Internet of Things," in A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium, Sydney, NSW, June 2014, pp. 1 - 10.
3. Roy T. Fielding et al. (1999, June) Hypertext Transfer Protocol – HTTP/1.1. [Online]. https://tools.ietf.org/html/rfc2616
4. E. Rescorla, T. Dierksv, and Inc RTFM, "The Transport Layer Security (TLS) Protocol Version 1.2," Internet Engineering Task Force
5. J. Höller et al., From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st ed.: Elsevier, 2014.
6. R. Khan, Univ. of Genova (UNIGE), Genova, Italy DITEN Dept., S.U Khan, R. Zaheer, and S. Khan, "Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges,"in Frontiers of Information Technology (FIT), 2012 10th International Conference, Islamabad, 2010, pp. 257 - 260.
7. Z. Shelby, ARM, K. Hartke, C. Bormann, and Universitaet Bremen TZI "The Constrained Application Protocol (CoAP)," Internet Engineering Task Force (IETF), Standards Track 2070-1721, June 2014. [Online]. https://tools.ietf.org/html/rfc7252.
8. C. Bormann, M. Ersue, and A. Keranen, "Terminology for Constrained Node Networks, "Internet Engineering Task Force (IETF), Informational 2070-1721, 2014.

