# Securing IoT Devices and SecurelyConnecting the Dots Using REST API and Middleware

Hittu Garg
Department of Computer Engineering
Nit Kurukshetra
Haryana, India
hittugarg6@gmail.com

Mayank Dave
Department of Computer Engineering
Nit Kurukshetra
Haryana, India
mdave@nitkkr.ac.in

***Abstract*—Internet of Things (IoT) is a fairly disruptive technology with inconceivable growth, impact, and capability. We present the role of REST API in the IoT Systems and some initial concepts of IoT, whose technology is able to record and count everything. We as well highlight the concept of middleware that connects these devices and cloud. The appearance of new IoT applications in the cloud has brought new threats to security and privacy of data. Therefore it is required to introduce a secure IoT system which doesn't allow attackers infiltration in the network through IoT devices and also to secure data in transit from IoT devices to cloud. We provide the details on how Representational State Transfer (REST) API allows to securely expose connected devices to applications on cloud and users. In the proposed model, middleware is primarily used to expose device data through REST and to hide details and act as an interface to the user to interact with sensor data.**

***Keywords—IoT,security, REST API, middleware.***

## I. INTRODUCTION

The Internet of Things interconnects computer devices integrated in everyday objects through the Internet, allowing them to send and receive data.There are two-fold advantages, we can empower our computers to gather information about surroundings without depending on humans and by processing the information collected we can reduce extravagance, loss, and cost.The Internet of Things allows for interaction between the physical world and the digital world.The digital world interacts with the physical world via sensors and actuators. These sensors collect information that must be stored and processed. Data processing can take place at the edge of the network or at a remote server or cloud.

The storage and processing capabilities of an IoT object are restricted by the resources available, which are constrained due to size limitation, energy, power, and computational capability. So these systems rely on IoT middleware to provide needed capabilities.

### *A. REST for machine-to-machine connectivity and IoT*

APIs allow to expose the connected device to users in a secure manner. RESTful APIs are widely being used in the modern web. Data transfer is usually done using JSON or XML over HTTP. It is a good model for the heterogeneous systems. REST API makes the device information easily available. They can standardize on a way to create, read, update, and delete this data. All these operations will be input to the REST query calls. REST APIs allow to delegate and manage authorization. The API can authenticate on the server and the server can authenticate to the API to prevent theman-in-the-middle attacks.

### *B. IoT Middleware architecture*

One way to handle such heterogeneous applications is that we can have a middleware platform that will become the bridge between things and applications in the cloud. Middleware packages and abstractshardware, and provides application programming interfaces (APIs) for communication, data processing, computing, privacy and security.

Figure 1 gives the overall picture of the role of middleware in I oT.In the broad category, there are four main components of an IoT System – things itself, the local network which can include a gateway, middleware, cloud (for user access control, Business Data Analysis etc.).

## II. SECURITY CHALLENGES IN IOT

IoT applications are becoming part of personal lives and data collected is rather sensitive and private to an individual. Privacy and Security issues must be addressed in all of these environments. As in the health sector, health data is highly critical for personal privacy, therefore it should not be accessed by an unauthorized entity.The recent voluminous DDoS attacks (October 2016) on DYN's servers that brought down many popular online services in the US, letting us know what can happen when attackers are able to leverage up to 150,000 unsecure IoT devices as malicious terminals [1]. To understand the overall approach to data security, there is a need to know about the security requirements for all key components of IoT system i.e. IoT devices, IoT users, Middleware/ IoT gateway, communication channel, and cloud applications.

### *A. Security Challenges in Constrained IoT Devices*

IoT systems are made of resource-constrained devices.

- Public key infrastructure is not suitable for IoT environments as it becomes a computationally expensive task to calculate ciphertext because of the large key size.
- Once the system is compromised, it is difficult to update it. In addition, shutting down a potentially compromised systems, reinstalling or restarting software, or replacing components or subsystems is not suitable for multiple IoT systems, as this can also result in severe business loss and disruption. It is also impractical and infeasible to use a secure firewall for IoT devices. [2].

978-1-7281-1253-4/19/$31.00 © 2019 IEEE

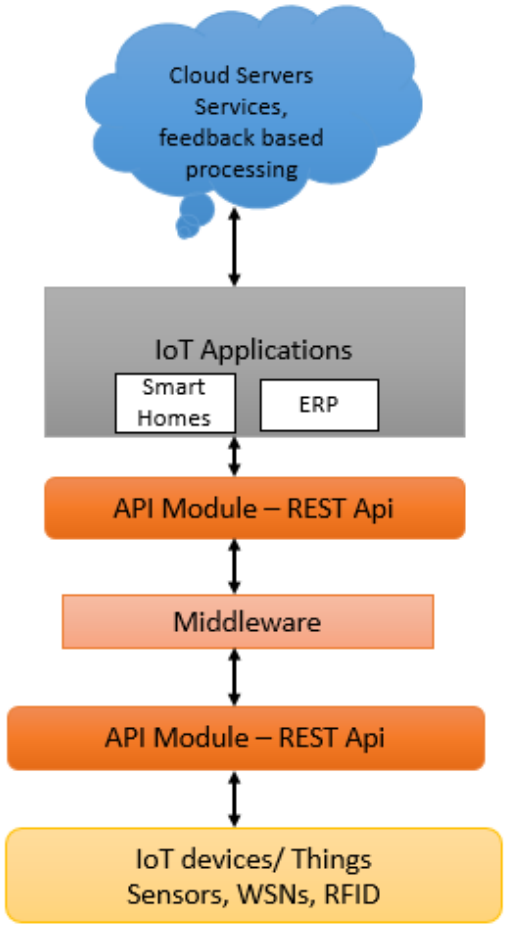


**Figure 1: IoT Middleware Architecture**

### *B. Need for Trusted Devices*

IoT middleware needs to manage a trust relationship with devices so that these devices can be authenticated and authorized to share data. It needs to enforce authentication prior to communication with any device enabling proof of the origin of data. These devices are assigned unique identitiesthat disallow the reuse of security credentials across devices.

### *C. Connecting IoT devices to Middleware.*

There is a requirement to connect a large number of heterogeneous smart devices like connected cars, connected wearables, smart cities etc. IoT devices are typically connected to the Internet through an IP stack. This stack now has its own complexity and requires a lot of power and memory from connected devices.

### *D. Security of Communication channel.*

IoT Data needs to be secured while at rest as well as while in transit to ensure data integrity. Security solutions are implemented in a way to detect unwanted intrusions and prevent malicious attacks on the communication layer. Also securing against attacks like Replay attacks, offline id guessing attacks, unauthorized login, user anonymity, and sensor node anonymity.Figure 2 gives an overview of security requirements for different pillars of IoT system.

## III. RELATED WORK

There are multiple solutions available to cater to different security requirements of an IoT System. The first and foremost requirement is mutual authentication between the IoT device and gateway within the resource-constrained environment of IoT System.

Recently, many authentication schemes for IoT have been proposed. In [3], the author suggested arobust anonymity preserving authentication protocol for IoT devicesthat provides mutual authentication between tag and reader through the server. This scheme uses Elliptic Curve Cryptography (ECC) to implement authentication.

It is already established that ECC based cryptographic algorithms are secure. The security of ECC relies on the difficulty of solving these two problems:-

- Elliptic Curve Discrete Logarithm (ECDL) problem: Let E be an Elliptic curve over a finite field. Let P and Q be the points in Zq (modulo q). It is difficult to compute the special integer α belongs to Zq satisfying Q = α P.
- Elliptic Curve DecisionalDiffie-Hellman (ECDDH) problem: aP, bP, and cP are three points in G. It is hard to verify if abP = cP.

As a method to provide the user the access of sensor or sensor data, the user is usually authenticated through the gateway. There will be mainly three parties: the user, gateway, and sensor. In [4] the authors proposes a verifiable, provable and privacy preserving user authentication scheme for wireless sensor networks (WSN). Authors point out that theHsieh and Leu's scheme [13] is not secure because of its several security shortcomings including Insider attack, off-line password guessing attack, user forgery attack, and sensor capture attack. For WSN, a new two-factor authentication scheme also based on ECC is presented. This scheme meets the authorization, confidentiality and integrity property for IoT security.

Different multi-factor authentication schemes are suggested for user authentication. Different factors like passwords, biometrics, and smartcards are used together for identity authentication mechanism. Like in [5], a three factor authentication scheme for WSN was proposed (2018). The three factors are password, smartcards and fingerprint identification. Here Gateway is supposed to be a trusted participant and bridge between user and sensor. This scheme successfully register a user and sensor on the gateway, authenticate the user to access sensor andprovides password changing facility tothe user.

Researchers have also proposed some ultra-lightweight authentication schemes which only involve simple bit-wise operations (like XOR, AND, OR, etc.) on tags. Therefore, it is very efficient in terms of storage and communication cost [6].

Razouk et. al. suggested a new Security Middleware Architecture based on Fog Computing and Cloud to support resource constrained devicesfor authentication [7]. IoT constrained devices communicate through proposed middleware which provides access to more computing power and enhanced capability to perform secure communications.

978-1-7281-1253-4/19/$31.00 © 2019 IEEE

This model is based on usual technologies like the "Constrained Application Protocol (CoAP)" and REST API for easy implementation.

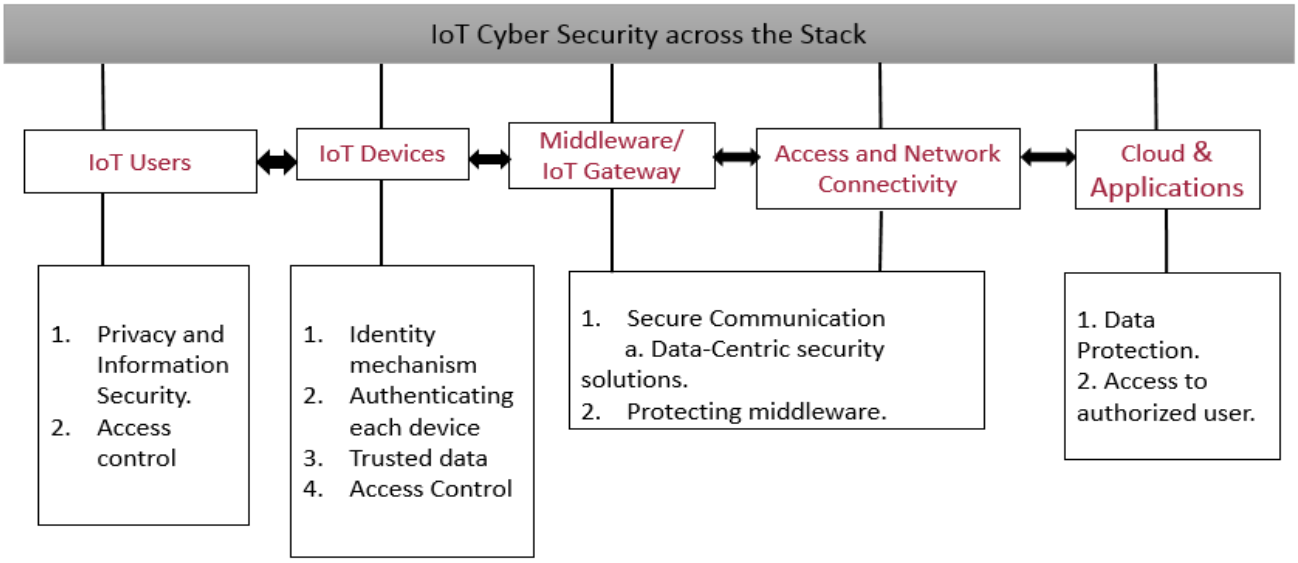


**Figure 2: Security Model for IoT System.**

## IV. Proposed Model

IoT devices typically connect to the Internet via an Internet Protocol (IP) stack. This stack is very complex and requires a lot of power and memory from the connecting IoT devices.In the proposed model IoT devices are connected locally through non-IP networks and are connected to the Internet through anintelligentgateway. This gateway act as an interface to the internet for the IoT devices. Because of the smart gateway devices can be hidden in enterprise behind multiple firewalls and will not require any inbound ports.

The proposed solution provides a middleware architecture that is able to adapt according to application requirements. Middleware is responsible for device registration, identification and database management. It also ensures the privacy and security of data. The stored data is exposed after authentication and authorization via REST API. Figure 5 summarizes the overall scenario.

There are many authorization protocols available. For example, OAuth is an open authorization protocol which can allow accessing the resource at middleware using username, password, and tokens. Figure 3 and 4 explainthe architecture to achieve this onboard flow.

Step 1:- Device Registration requires an Authorized user who already has created an online account through middleware.

Step 2:- Whenever an authentication request comes from the IoT device, gateway validates the devices request including the device payload by accessing the exposed REST API. Now the gateway will in turn request access to exposed API with its own credentials. The request will contain gateway id and secret key as input parameters. The API will both authenticate and authorize the gateway and will validate the request. All methods in REST API requires authentication. Once the gateway is authorized, a response in an encrypted form containing device details is sent back to the gateway. Now after verifying, gateway gives the access token to the device and device can now send real-time data to the gateway.

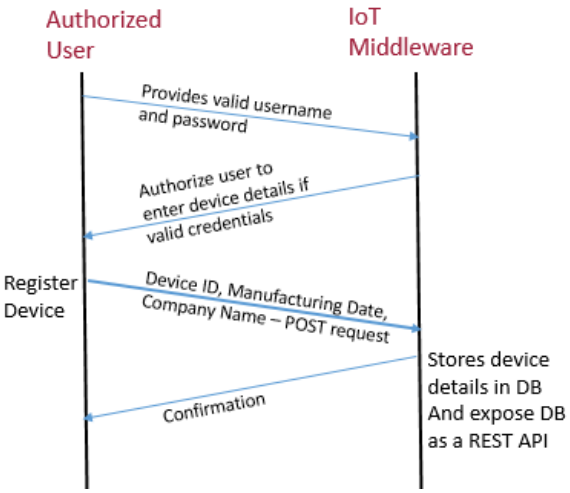


**Figure 3: Device Registration**

978-1-7281-1253-4/19/$31.00 © 2019 IEEE

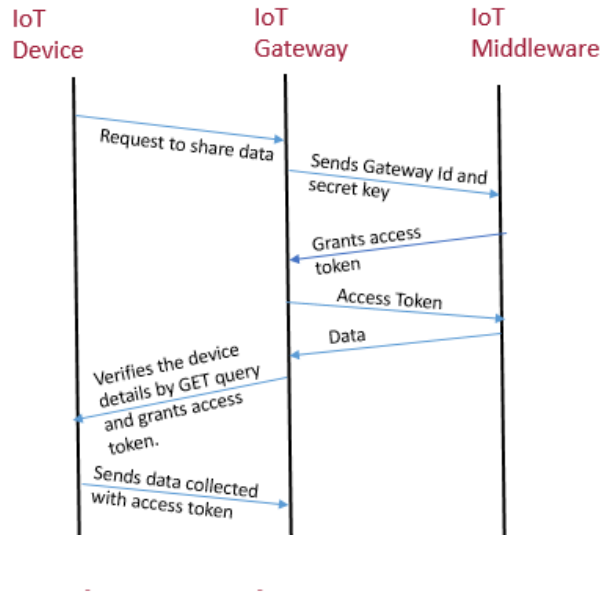


**Figure 4: Data sharing**

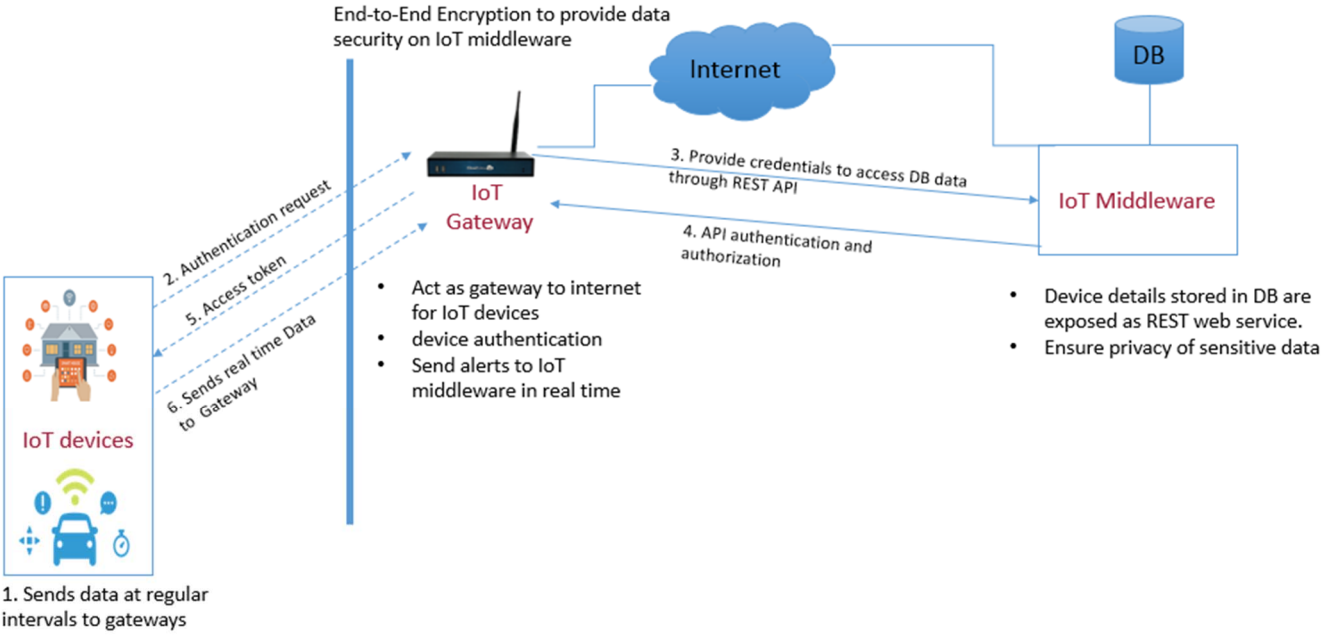


**Figure 5:Proposed System Model.**

*A. Defining required REST endpoints.*

The real need in an Internet of Things environment is to move data efficiently, quickly and securely. An API makes this easier and thus becomes the core of IoT. A RESTful web service uses HTTP for machine to machine communication, also for transferring machine readable file formats such as XML and JSON. We have defined different REST endpoints for different operations.

*B. Accessing REST endpoints*

IoT REST API-Inbound is enabled with authentication and authorization. After properly authenticating the gateway operations, the gateway gains the privileges to access the URLs. Now gateway will build a jQuery client that consumes a RESTful Web Service. All the service requests will be consumed at a URL. The service will respond with a JSON representation of device details.

- REST resource for getting Device Details:-

  a. Retrieve access token to access API-inbound at middleware.

| | |
|---|---|
| REST URL (Access token URL) | https://middleware-0ff5f6bf8404dsdca4012sa1c0426f689905.identity.c9dev1.oc9qadev.com:443/oauth2/v1/gateway-id |
| Request | POST |
| Response | {<br>“access_token”:” sdjdwkwsjdjnjsncjdcnhwdnjsnjnjdncjddjiwdjjdnjndjcxmsmcxsncwenddscdcnwdeidmd |

978-1-7281-1253-4/19/$31.00 © 2019 IEEE

| | |
|---|---|
| | mlkwkqnjqwdnjwdndkwefjuiefjrnfjfnjefhhyrfbrhfb",<br>"token-type":"bearer",<br>"expires_in":"3600"<br>} |
| Mandatory fields | Gateway-id, secret key |

b. To access device details after authorization

Figure 6 shows the sample code to hit the REST API. The given module is a simple JavaScript function that uses jQuery's $.ajax () method to use REST service at a specified URL. If successful, it will assign JSON received to the variable data.

| | |
|---|---|
| REST URL | http://localhost:8080/test/webresources/com.mycompany.test.devicedetails/{id} |
| Request | GET (application/ json) |
| Response<br>In JSON | {<br>"deviceIdentifier":"a","<br>deviceManufacturingDate":"2018-12-16T18:30:00Z[UTC]",<br>"uid":"abcd12344"<br>} |
| Mandatory fields | Device Identifier |

```
$.ajax({
        type: "POST",
    url:  "http://localhost:8080/test/webresources
/com.mycompany.test.devicedetails",
    data: JSON.stringify({
    deviceidentifier : self.newDeviceId(),
    devicemanufacturingdate : self.newMDate(),
    uid : self.newUid()
        }),
    headers: {
    'Content-Type': 'application/json'
        },
    success: function() {
            console.log(data);
        },
    error: function(err) {
            console.log("AJAX Error: " + err);
        }
      });
```

**Figure 6: Sample AJAX call to hit the REST API through a URL**

## V. ANALYSIS OF PROPOSED MODEL

In this paper, we proposed a secure IoT framework that ensures end-to-end security from IoT application to IoT Devices. We can evaluate the security of the system by analyzing the integrity of each component.

- IoT devices are isolated and have no interaction with the outside world. They are connected to a gateway and this gateway act as an interface to the internet for the IoT devices. Because of the smart gateway devices can be hidden in enterprise behind multiple firewalls and will not require any inbound ports. Therefore there are no chances of compromising these devices by an attacker.
- IoT gateway acts an intermediate between IoT devices and middleware. Gateways are enabled to call REST API and exchange all information securely.
- Communication between IoT gateway and middleware is secure with traditional cryptographic algorithms. There is no need to use lightweight algorithms because the two parties involved are not resource constrained.
- Authentication and Authorization is taken care by REST APIs which makes the whole process less complex and compatible with industry standards.

## VI. CONCLUSION

We proposeda middleware architecture which provides an end-to-end security solution for contributors who upload sensing data. This approach allows an end to end encryption of data to secure data in transit. In the proposed middleware solution all IoT system constraints are taken into consideration. REST API is used for communication and data exchange. Middleware successfully assists IoT development by exposing REST API and providing an interface to the user to register their IoT devices and then securely accessing data collected by the device.

## VII. FUTURE SCOPE

Our future plan is to provide a multi-factor user authentication scheme at middleware to provide secure access to sensory data to an authorized user. Also to provide a privacy solution to prevent any data leakage and data breaches at middleware.

## REFERENCES


[1] Kolias, C., Kambourakis, G., Stavrou, A., & Voas, J. (2017). DDoS in the IoT: Mirai and other botnets. Computer, 50(7), 80-84.

[2] Alrawais, A., Alhothaily, A., Hu, C., & Cheng, X. (2017). Fog computing for the internet of things: Security and privacy issues. IEEE Internet Computing, 21(2), 34-42.

[3] Tewari, A., & Gupta, B. B. (2018, January). A robust anonymity preserving authentication protocol for IoT devices. In Consumer Electronics (ICCE), 2018 IEEE International Conference on (pp. 1-5). IEEE.

[4] Wu, F., Xu, L., Kumari, S., & Li, X. (2017). "A privacy-preserving and provable user authentication scheme for wireless sensor networks


978-1-7281-1253-4/19/$31.00 © 2019 IEEE

based on internet of things security". Journal of Ambient Intelligence and Humanized Computing, 8(1), 101-116.

[5] Li, X., Niu, J., Kumari, S., Wu, F., Sangaiah, A. K., & Choo, K. K. R. (2018). "A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments". Journal of Network and Computer Applications, 103, 194-204.

[6] Chien, H. Y. (2007). SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity. IEEE transactions on dependable and secure computing, 4(4), 337-340.

[7] Razouk, W., Sgandurra, D., & Sakurai, K. (2017, October). "A new security middleware architecture based on fog computing and cloud to support IoT constrained devices".In Proceedings of the 1st International Conference on Internet of Things and Machine Learning (p. 35). ACM.

[8] "REST API for Oracle Internet of Things Cloud Service", docs.oracle.com/en/cloud/paas/iot-cloud/iotrq/QuickStart.html.

[9] Sfar, A. R., Natalizio, E., Challal, Y., & Chtourou, Z. (2018). "A roadmap for security challenges in the Internet of Things. Digital Communications and Networks", 4(2), 118-1375.

[10] Fremantle, Paul & Scott, Philip. (2015). A security survey of middleware for the Internet of Things. 10.7287/PEERJ.PREPRINTS.1241

[11] Ayoade, G., El-Ghamry, A., Karande, V., Khan, L., Alrahmawy, M., & Rashad, M. Z. (2018). Secure data processing for IoT middleware systems. The Journal of Supercomputing, 1-26.

[12] He, D., Chen, J., & Zhang, R. (2012). An efficient and provably‐secure certificateless signature scheme without bilinear pairings. International Journal of Communication Systems, 25(11), 1432-1442.

[13] Hsieh, W. B., & Leu, J. S. (2014). A Robust ser Authentication Scheme sing Dynamic Identity in Wireless Sensor Networks. Wireless personal communications, 77(2), 979-989.

[14] Choi, Y., Lee, D., Kim, J., Jung, J., Nam, J., & Won, D. (2014). Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography. Sensors, 14(6), 10081-10106.

978-1-7281-1253-4/19/$31.00 © 2019 IEEE