International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 03 | Mar 2020 www.irjet.net p-ISSN: 2395-0072

Effective Authentication of Medical IoT Devices using Authentication Server

SHALINI A[1], ARIGANESH T[2], ARJUN GANESH S[3], VIJAY BALAJI G P[4]

[1]Assistant Professor, Department of Computer Science and Engineering, Kingston Engineering College, Tamilnadu, India.
[2,3,4]UG Scholar, Department of Computer Science Engineering, Kingston Engineering College, Tamilnadu, India.

Abstract - *In the scenario of Internet of things used in medical appliances, the data from each device cannot be sent directly to the database. This technique is very much vulnerable to man-in-the-middle attack and replay attack. To prevent these kind of vulnerabilities and to enhance the security measures, we are implementing this new kind of technique. So there is a need for improving the security of the components so that only authorized personnel are only allowed to access the devices. Here the Kerberos server will be acting as the hub for the IoT devices. This server is responsible for the authentication and IoT device management. The Elliptic Curve Digital Signature Algorithm (ECDSA) is used for authentication and encryption to reduce the key size. Here we are not giving a session for each devices on authentication because, there may be the chance of data forging after the authentication by the same microcontroller. Since the medical data is quite sensitive, each data will be validated with the digital signature.*

Key Words: IoT, Cyber security, ECDSA, Kerberos, Authentication

1. INTRODUCTION

Recently the deployment of IoT devices has grown exponentially, and it's obvious that several recent cyber-attacks are IoT-enabled. The attacker initially exploits some vulnerable IoT technology as a primary step toward compromising a critical system that's connected, in how, with the IoT. for some sectors, like industry, smart grids, transportation, and medical services, the importance of such attacks is obvious, since IoT technologies are a component of critical back-end systems. However, in sectors where IoT is usually at the end-user side, like smart homes, such attacks are going to be underestimated, since not all possible attack paths are examined.

Advanced information services can be developed using IoT that needs real-time data processing and requires data centres of high storage and computing power. But, this will introduce many security problems since attack surface, complexity, heterogeneity and number of resources keep increasing.

Now-a-days, connecting Medical devices and equipment to internet are increasingly adopted. Life of the patients have been altered using these devices. Without the need of a doctor, health statistics can be gauged at home has improved the wellbeing.

Introduction of malware into the equipment or by unauthorized access to data and configuration setting of the devices are the ways of cyberattacks in IoT. They are not based only on device, but also on the networks in which they are connected.

Health care industries are becoming more prone to cyberattacks. Increase in number of networked medical devices creates urgency for makers of these devices to understand and mitigate threats to their devices' security. More and more medical devices containing embedded systems keeps increasing makes it vulnerable to security breaches and affect the device operation.

Using patient's health data, phishing schemes are sent or identity can be stolen when combined with mined data. Discriminating device maker's safety and efficacy can be done using product performance data which can also be sold to competitors.

To prevent attackers from accessing the information on the devices, instead of sending the information directly, we are using Authentication Server to verify the identity of the doctors and patients. Using this server, only authorized personnel will be able to view the devices and we set privileges for each user so that only a selected few can modify the instructions inside the device.

2. Existing System

In current scenario the IoT nodes are connected in a wireless un-reliable environment. The data is shared between the nodes are sent directly without any means of security measures. Also the data that is stored on the database does not have access privileges. Since the IoT devices can be accessed through Internet there is a constant risk of being attacked by hackers. If supposedly an attack took place, there is no way of knowing the identity of the attacker.

Any foreign device can connect into the network without any permission and starts to send and receive data

between the nodes. In medical field the data of the patients is to kept secret to avoid any data breach, but accessing the data from the devices is very straight forward and requires no authentication.

Disadvantages:

- Misuse of the devices.
- Device Failure.
- Data and Identity theft.
- No Authentication or Encryption.
- Difficulty in detecting and recovering from attacks.

2.1 Proposed System

To prevent these security issues in IoT, we are using a Kerberos like server which is going to act as a hub. All the IoT nodes will be connected via intranet to the hub instead of getting connected directly to the online database. Each IoT nodes will concatenate the date and time along with the sensor data and encrypt it with the Elliptic Curve Digital Signature Algorithm. These encrypted data will be sent to the Kerberos via intranet and the Kerberos will decrypt the data and match the digital signature. If the digital signature matches, the data will be uploaded to the online database, else it will be omitted. The values can be read in form of a graph for better view. And we also give privileges for each user so that only selected few can change the state of the device or modify the information stored in it.

Advantages:

- Easier End to end Authentication.
- Avoiding Man in the middle attacks.
- Secure transfer of data using encryption.
- Ability to set privileges to each user.
- Latency is reduced by using Intranet.

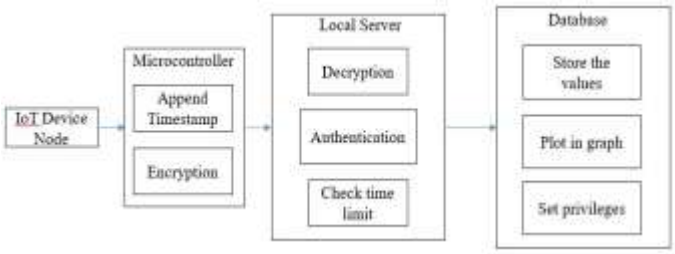

Fig -1: Architecture Diagram of proposed system

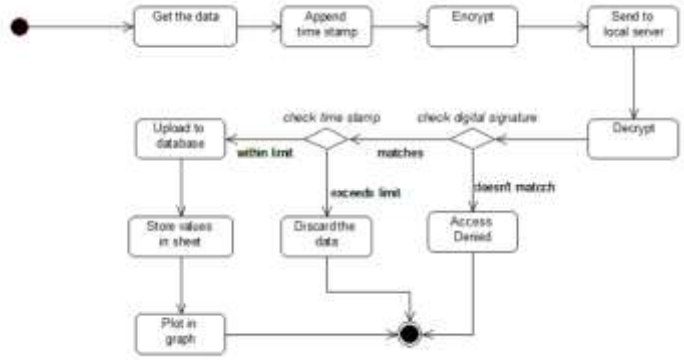

Fig-2: Dataflow Diagram

3. MODULES

3.1List of modules

- Data Collection Module.
- Local Server Module.
- Online Storage Module.

3.2MODULES DESCRIPTION

Data Collection Module

This module comprises of using a medical device such as Insulin pens, Temperature monitor, Ingestible sensor for collecting the data. NodeMCU Microcontroller is connected to the devices. The NodeMCU is programmed with Arduino Kit to accept and encrypt the data from device. The microcontroller first concatenates the data with the timestamp and then encrypts it and sends it to the local server.

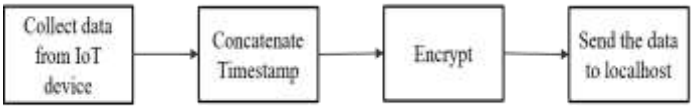

Fig-3: Collecting data from NodeMCU

Local Server Module (Kerberos)

The Local Server receives the encrypted data, decrypts it and checks the digital signature, timestamp and uploads it to the cloud database. This local server is written in Flask Framework and written in APIs. Based on the received data the Local server decides whether to send it to online database or drop it. A Time limit is set for data to be received in local host. If the comparison of sent and received time exceeds the given limit, then the data is dropped. The data that is sent successfully can be viewed in the server in form of graph.

Fig -4: Verification of data in local server

Online Storage Module

This module consists of an online cloud database software to store the data from the Local server. Data that is successfully received is stored in the sheet book format in the database. And the sheet is constantly updated when new data comes into the place. A graph is also set for ease of viewing the data. Access permissions can also be set for individual users so that only authorized ones are allowed to view and access the data.

ig-5: Storing in database

4. ALGORITHM

Elliptic Curve Digital Signature Algorithm

The Elliptic Curve Digital Signature Algorithm provides a new way of implementing Digital Signature Algorithm (DSA) which uses Elliptic Curve Cryptography. In elliptic curve cryptography the size of the public key in size of bits is twice of its security level in bits.

Generation Algorithm:

- Both sender and receivers agree on curve parameters.
- Calculate the hash function of the message.
- Select the left most bits of the hashed message.
- Choose a secure random number.
- Calculate a curve point using the random number.
- Using curve points generate the signature co-ordinates.

Verification Algorithm:

- Either sides should have the copy of public key curve points.
- Check if the curve point is equal to identity element, then the key is valid.
- Verify that signature co-ordinates are in the range of given order,
- Calculate hash value using the same function used in generation.
- Select the left most bits of the hashed value.
- Calculate the new curve points using curve's base point and public key.
- If the new curve points match the generated ones, then the signature is valid.

Diffie-Hellman key exchange

Diffie-Hellman key exchange was one of the first public key protocols for securely exchanging cryptographic keys over a public channel. This method allows to establish a shared secret key between two parties that have no prior knowledge of each other over an insecure channel.

Algorithm:

- First, both sides agree on base and modulus which are public.
- Then, either side chooses an integer that is known to that side only.
- Using this secret integer, public keys are calculated for both sides by using base and modulus.
- When public keys of each other are known, they use it to deduce the shared secret key known only to both sides.

5. CONCLUSION

Here we conclude that, in this project we have created a more secure and effective way to provide authentication to medical IoT devices in an un-reliable environment. There are three phases in this paper: Collection phase, Authentication phase, Storage phase. In Phase 1 the micro controller collects the data from node, appends a timestamp, encrypts it and sends to Local server. Phase 2 comprises of decrypting, authenticating, checking the time limit and uploading to online database. Storing and giving privileges happens in Phase 3. Thus, providing more security and authentication so that risk of attacks like Man in the Middle attack is reduced. This can be further improved by using privately owned servers and database.

REFERENCES

- Caiming Liu; Yan Zhang; Zhonghua Li; Jiandong Zhang "Dynamic Defense Architecture for the Security of the Internet of Things", 2015.
- Lilla Nagy; Adrian Colea "Router-based IoT Security using Raspberry Pi", 2019.
- Danish Showkat; Shubranil Som; Sunil Kumar Khatri; Armaan Singh Ahluwalia "Security Implications in IoT using Authentication and Access Control", 2018.
- Guojun Ma "A Security Routing Protocol for Internet of Things Based on RPL", 2017.
- Ayman El Hajjar, "Securing the Internet of Things Devices Using Pre-Distributed Keys", 2016.
- Seungyong Yoon; Jeongnyeo Kim "Remote security management server for IoT devices", 2017.
- Boheung Chung; Jeongyeo Kim; Youngsung Jeon "On-demand security configuration for IoT devices", 2016.
- Joseph Bugeja, Bahtijar Vogel; Andreas Jacobsson; Rimpu Varshney "IoTSM: An End-to-end Security Model for IoT Ecosystems", 2019.
- J M Blythe; S D Johnson "The consumer security index for IoT: A protocol for developing an index to improve consumer decision making and to incentivize greater security provision in IoT devices", 2019.
- Rishav Gauniyal; Sarika Jain "IoT Security in Wireless Devices", 2019.