Design Engineering

ISSN: 0011-9342 | Year 2021
Issue: 9 | Pages: 623 - 640

Secure Data Communication in Iot Supported Social Networking System Using REST API and JSON Web Token

J. Auxily Jovita
Research Scholar
Division of Computer and Information Science,
Annamalai University,
Annamalainagar – 608002
Email: Auxilyantony@Gmail.Com

Dr. G. Ramachandran
Associate Professor,
Dept. of Computer Science & Engineering,
Annamalai University
Annamalainagar – 608 002.
Email: Gmrama1975@Gmail.Com

Dr. N. Edison Rathinam
Assistant Professor,
Dept. of Computer Science,
K.C.S. Kasi Nadar College of Arts and Science
R.K. Nagar,Chennai – 600021
Email: Edisonrathinam@Gmail.Com

ABSTRACT

Today, Internet of Things (IoT) is facing various problems due to the incredible development. In this paper the importance of REST API is analyzed and it is employed with JSON. Different kinds of threats related to safety and confidentiality of data are familiar. To resolve the data preaches, secure communication is required for IoT devices as well as IoT system. REST API is utilized for ensuring and enhancing security of IoT system in this paper. End to end encryption is ensured making use of IoT middleware/gateway technology. Mutual authentication among IoT devices is implemented by incorporating authentication and authorization to the IoT system. Real time data communication is significantly concentrated via IoT gateway. Further privacy and confidentiality of data are focused using JSON token. Thus proposed model is working with all the essential security features of IoT system and combat against various attacks like impersonation attack, forgery attack, offline password guessing attack, replay attack and others.

Key words – REST API, JSON, security, IoT

I. INTRODUCTION

Data communication of IoT system integrates electronic devices like computers and other gadgets with internet. Data collection is accomplished without any human intervention which reduces profligacy, data loss and expenditure. IoT acts as an interface amid physical and cyber world. Sensors and actuators are utilized for ensuring such interactions. Information collected by sensors which are accumulated for processing [1]-[2]. Data processing is implemented in the end of network or remote server. Memory and processing potentials of IoT system depend on the accessible resources in terms of size, power, control and computational ability. Middleware technologies are incorporated for accomplishing those limitations [3]-[5]. General representation of IoT system is presented in fig.1.

APIs provides secure connection between the ends. Current cyber world utilizes RESTful APIs majorly. JSON is utilized for transferring data over HTTP [6]-[7]. Further they are providing well support for heterogeneous environments. It is easy to get available device information while using REST API and data management operations (i.e. creation, reading, updating and deletion) are normalized by them. Every operation is set as input for REST query calls. Authorization is ensured and handled making use of REST APIs. Thus API validates over server and API is authenticated by the server by enabling protection from man in the middle attacks.

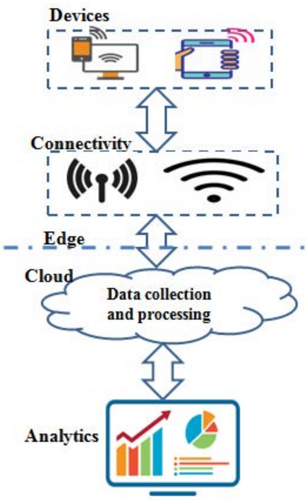


Fig.1 General representation of IoT system

IoT middleware acts as an interface amid applications and allows managing heterogeneous system. Hardware representation is summarized by middleware. In addition middleware affords APIs for various purposes such as messaging, computing, data processing, confidentiality and safety. IoT system is comprised with four primary components which are IoT devices, host network associated with a gateway, middleware and cloud environment [8]-[9]. In such way middleware contributes to promote multiple applications like user access control, Business Data Analytics and so on.

Data collection is an key process of IoT applications in which private and confidential information of .every person is intended. It is mandatory to concentrate on the constraints related to safety and confidentiality. Illegitimate entities will not be allowed to get personal private data especially related to patient's health. In October 2016, DYN's servers of US got hugely affected by DDoS attacks in which 150,000 IoT devices are maliciously infected [10]-[11]. Data security of IoT system (i.e. IoT devices and users, IoT Gateways, communication medium and IoT applications) is recognized by meeting all the security requirements.

Public key crypto system is not appropriate for IoT system because of its computational cost towards large key sized cipher text computation. Updating system is complicated when it is affected by malicious attackers. Further compromised systems are unable to shut down. Software restart and components replacement are also difficult for compromised IoT systems which will lead to business disorder. Utilization of firewalls can't too ensure sufficient security solution of IoT devices [12].

Trust management amid IoT devices is implemented by middleware which ensures authentication and authorization of shared data by IoT devices. Source of data is focused for authenticated communication in between devices in IoT system. At every time of access, unique security credentials are allocated to every device and they are reused.

Internet and IP stack are main contributors while connecting huge amount of heterogeneous smart devices such as connected automobiles, connected wearable's, smart cities, and smart homes and so on. Thus it becomes complicated in terms of power consumption and memory storage.

Data integrity should be maintained while handling IoT data in secured manner. For detecting unnecessary intrusions and preventing vulnerabilities in communication layer, security remedies should be provided effectively. Further security is required against various attacks like replay, offline id or password guessing, impersonation, anonymity of user and sensor nodes and so on. Security requirements that should be satisfied for IoT system is depicted in fig.2.

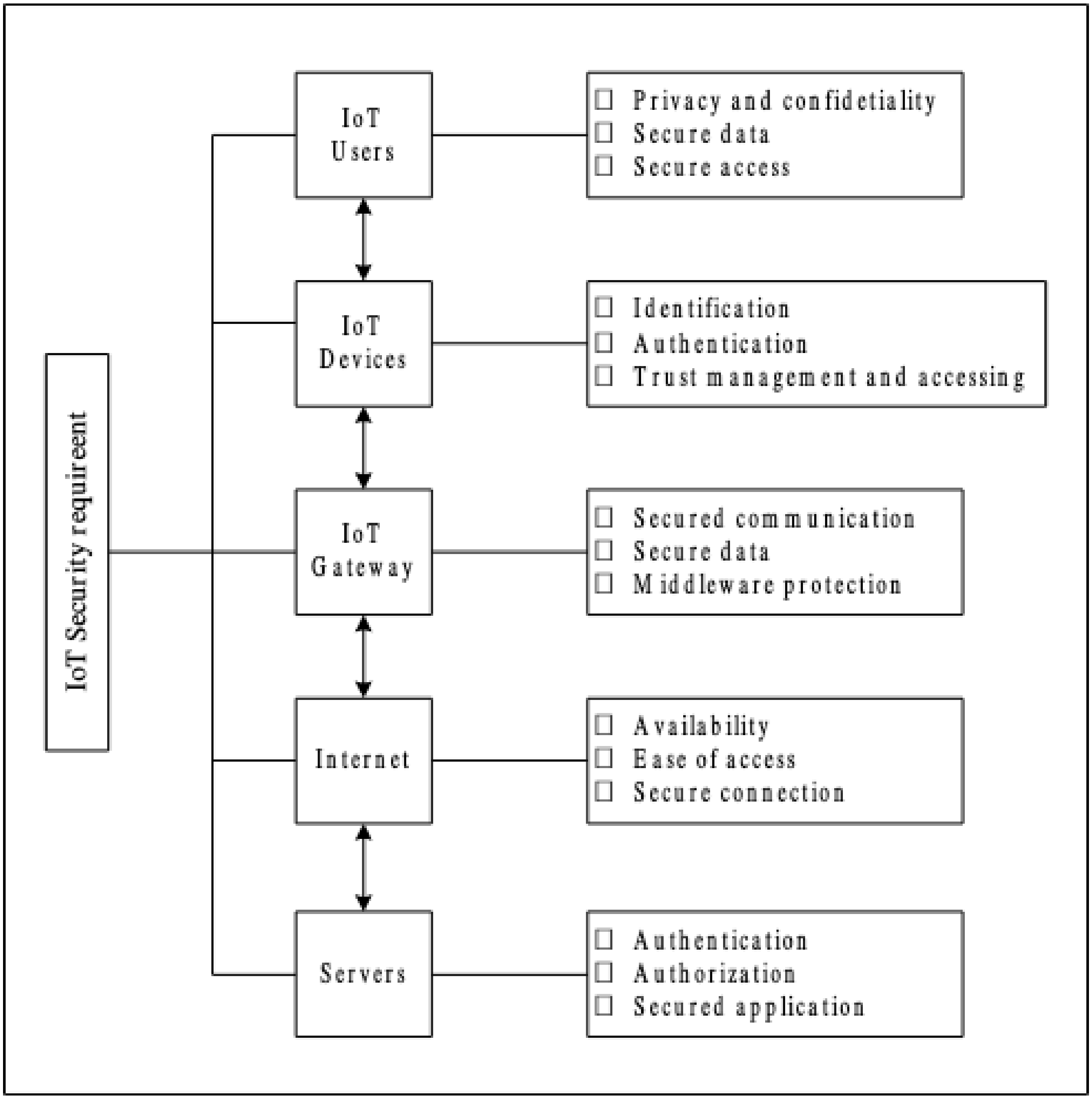


Fig.2 IoT system's Security model

The rest of the paper is classified as the following sections: Section II describes the previous research works with respect to the IoT system and the security features related to it, section III details about REST API and JSON token and their importance, section IV describes the prototype of proposed model and its working procedures in detail, and finally conclusion is provided.

II. RELATED WORK

Various solutions are provided for IoT system in order to meet distinct security requirements. Mutual authentication is considered as one of the primary needs amid devices and gateway for the entire IoT system. Various authentication schemes are proposed with the same intention. Multiple research works are carried regarding ultra lightweight validation schemes which focused over simple bit wise operations on tags such as XOR, AND, OR, and so on. They seem effective in terms of memory and cost [13]-[16].

Cassar et al. presented semantic service to support IoT device matchmaking technique which is inherited from probabilistic service matchmaking [17]. It is assured that this hybrid method will conquer the issues while describing semantic service. Multiple sensors are used by Boman et al. for displaying sensor networks which is inter related with GSN and Firebase systems [18]. For describing sensor node and its characteristics, XML is utilized with various remedies regarding problems.

Wu et al presented a wireless sensor networks (WSN) based authentication scheme aiming to provide protection against privacy and authentication vulnerabilities [19]. Service Oriented Architecture (SOA) utilized URI in the research work of Zhang et al publish and subscribe model of IoT is concentrated [20]. During this service implementation, identification of device is required. XML service description is customized to accomplish identification.

Cryptanalysis is performed at Hsieh and Leu's scheme [21] and they found possibilities of vulnerabilities against various attacks like insider attack, impersonation, offline password guessing and sensor capturing. To resolve that security illness ECC based two factor authentication schemes is implemented with the goals of satisfying the security requirements of IoT such as the authorization, privacy and reliability. Souza et al. implemented set of vocabulary in order to determine IoT service [22]. Uniform Resource Identifier (URI) is utilized for IoT identification like the similar representation of web and semantic web. XML format is implemented based on the syntax of URI. Further, Web Services Description Language (WSDL) is described in favor of IoT device with the help of URI.

Hur et al. considers interoperability as a primary problem amid IoT system and distinct platforms [23]. Idea of Semantic Service Description (SSD) ontology is implemented which provides semantic representation of diversity between multiple platforms. JSON-LD (Linked Data) is utilized for accumulating semantic meta data. For recovering that data, Web Linking and POWDER described by property are utilized together [27]. SSD ontology is comprised with three primary ideas which are Property, Capability and Server Profile. Every idea is intended for maintaining interoperability amid physical things and platforms. Device configuration details are described by server profile such as HTTP methods, server URI, API keys, and so on. IoT device is assumed with the development of complete ISO/OSI stack model and internet connectivity. TCP/IP protocol play key role for supporting these services.

Similarly same researchers concentrated on the advantages of previous proposal and utilized conventional web technologies [24]. Differences in every IoT system are identified on the basis of structure and format of data and APIs. In addition those differences will affect interoperability and data discrepancy. Needs of automatic employment is examined especially Topology and

Orchestration Specification for Cloud Applications (TOSCA) which is intended for IT services illustration. Further TOSCA is applicable for deploying services in IoT system. During experimentation, processors of 5 IoT devices utilized intended in the publication of semantic meta data over internet. Based on the results their proposed approach can be applied with several IoT devices over multiple platforms.

Ngu et al categorized the distinct design patterns of IoT middleware and analyzed further on the basis of composition, flexibility and safety [25]. IoT service detection is highlighted as a major problem by authors. To resolve the issues they proposed Service Xplorer which is a non ontological method and also acts as search engine meant for heterogeneous IoT device and services. Search engine utilizes WSDL for service description. One more ontological method is also proposed by them which is relate to data analytics in the usage of IoT devices. Correlation among devices and services is created. Load balance is accomplished with the help of those relationships. Safety and confidentiality related issues are also analyzed. In such way proposed approach seems as light weighted with the utilization of minimum bandwidth, CPU and resources. Public key cryptographic approaches like NTRU, ECC and AE are employed. In addition they found that IoT middleware should be implemented with essential security features.

PalCom Object Notation (PON) is employed by Nordahl and Magnusson which is a light weighted data exchange format meant for IoT applications [26]. It is basically a middleware JAVA framework which makes simpler of generating dynamic networks amid device and IoT system. Features of JSON like structure and trimness are mixed together to describe the structured data's format by PON. FORTRAN based String representation (i.e <Length><Type><Data>) is utilized to enhance data handling such as 5s IOTIF. PON is basically following textual representation of data. Proposed implementation is applied in healthcare system without failure. After testing, results depicts that PON is 30% lesser than JSON and 70% better than GSON.

Razouk et. al. proposed a latest Security Middleware Architecture on the basis of Fog and Cloud Computing which mainly focused on authentication of resource-constrained devices [28]. It provides added computation power and enhanced secure communication for IoT devices. Tewari et al proposed an IoT based authentication protocol for ensuring robust anonymity protection which affords mutual authentication among tag and reader via server [29]. Elliptic Curve Cryptography (ECC) is utilized mainly for authentication as many researchers have proven it's effective in security. ECC's security is basically on resolving two distinct problems which are Elliptic Curve Discrete Logarithm (ECDL) problem and Elliptic Curve Decisional Diffie-Hellman (ECDDH) problem. User, gateway and sensor are the key components of this system in which sensor and data are accessed by user and authentication to user is provided via gateway.

Various authentication schemes are implemented on the basis of multiple factors to authenticate users. Multiple credentials such as smartcards, biometrics and passwords are utilized collectively for better authentication. Further three factor authentication scheme is proposed for WSN with the consideration of the credentials like smart cards, biometrics and password [30]. Gateway acts as third party trustworthy entity amid user and sensor. Successful registration of user and sensor over gateway is the initial step. Further user's authentication is verified at every access of sensor making use of unique password.

Middleware architecture is proposed for ensuring end to end security over sensory data [31]. In order to exchange data between the ends REST API is employed which enhances communication in secured manner. Data collection is performed securely with the implementation of REST API as a interface among them. It is applicable to the IoT system and cloud environment.

JSON Web Token (JWT) is utilized for managing authentication requests among the server and database [32]-[35]. This proposed approach generates JWTs for very client entity on the basis of random timestamp values which enhances legitimacy of client over server. It mainly concentrates on security and authentication. Fast creation of new tokens lead to *O(1)* time complexity over server in which attacker cannot find client's signature. Unauthorized requests are restricted while using this technique. Data collection service of Multi-LPWAN (multiple low-power wide-area networks) technology is concentrated in the aspects of development and accomplishment [36]. Collector or provider service is related to IoT device. Data is created from various resources and combined as a single source of REST API. Data integrity is mainly focused.

III. IMPORTANCE OF REST API AND JSON WEB TOKEN

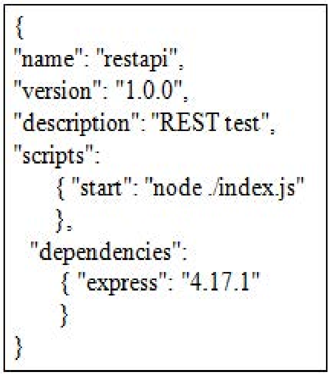
In this section importance of REST API and JSON web token are explained in detail.

A. REST API

REST (REpresentational State Transfer) is a structural design developed by scientist Roy Fielding which affords standards amid computers and web [37]. It makes communication simpler within the network. They are also represented as RESTful systems. Systems belong to REST standard are stateless i.e. client and server don't aware the states of each other. It mainly concentrates on resources. Client request of REST contains an HTTP verbs (GET, POST, PUT and delete), a header, resource path and message body which is optional.

REST API is also defined as RESTful API which is an application programming interface that matches constraints of structural design of REST for interacting with web services. To construct and integrate application software, stack of protocols are designed by API which is assumed as agreement amid user and provider of information. Connection establishment is referred as call and content is sent as response. API is helpful while interacting with a system for retrieving information and it makes communication for defining the request. REST API acts as interface amid users and web services and it implements various advantages. It ensures enhanced resource sharing in secure and authenticated manner.

While connection is established through RESTful API, client request is created. Various formats such as JSON, XLT, HTML, PHP, Python and plain text are implemented through HTTP. HTTP methods are important for every HTTP request associated with RESTful API. Request information includes meta data, URI, cookies, caching, authorization and so on. Every HTTwP connection is comprised with request and response headers and status codes. Sample REST API is provided in the below fig.3.



```
{
"name": "restapi",
"version": "1.0.0",
"description": "REST test",
"scripts":
      { "start": "node ./index.js"
      },
  "dependencies":
      { "express": "4.17.1"
      }
}
```

Fig.3 Sample REST API

An API is treated as RESTful with the following conditions.

□ It should be implemented in client server representation and resources and requests are controlled via HTTP.

□ Stateless communication should be maintained between the client and server.

□ Data is interacted in the format of catching

□ Standardized interface amid components is transferred and it requires requested resources, self descriptive messages, hyper text or hyper media, layered representation and resources constrained.

Advantages of JWT

□ It supports various languages and environments

□ It enables communication among multiple web applications

□ Client requests are handled independently

□ It is light weighted and supports scalability for architectural design

B. JSON web token

JWT (JSON Web Token) is a RFC 7519 open standard which describes secure transformation of messages in among the ends with the representation of JSON object [38]. Digital signatures are used for message verification. Various cryptographic algorithms are utilized for secure messaging. HMAC (Hash-based Message Authentication Code) is utilized by JWTs for signing with secret value. Likewise JWTs sign with a pair of public or private keys making use of algorithms like RSA (Rivest–Shamir–Adleman) and ECDSA *(*Elliptic Curve Digital Signature Algorithm).

JWTs use signed tokens in order to maintain secrecy among the ends other than encrypted JWTs. Integrity of shared data is verified while using signed tokens which results better than encrypted tokens. While signing with a pair of public or private keys, signature provides certificates to the ends which are provided with private keys. Need of JWTs are listed as following scenarios

□ **Authorization** is one of the important scenarios with the usage of JWT. At every login of user, following request will comprise JWT which permits user for accessing respective routes, services and resources allowed with token. JWT implements the effective security feature Single Sign On due to low overhead and interoperability among various platforms.

□ **Information exchange** is carried in efficient way by JWTs securely between the ends. Signature verification plays significant role where signature is computed by utilizing the header and payload of JWTs.

□ Structure of JWT is classified into three parts which are header, payload and signature and they are segregated by dots (.) (i.e xxxx.yyyy.zzzz). Header is comprised of two parts which are type of token (usually JWT) and algorithm used for signing (such RSA, SHA256 etc..). Payload is contained with claims. Claims are defined as entity representing statements. Registered, public and private are three distinct types of claims. Registered claims are already defined and they are not compulsory. Examples of registered claims are iss, exp, sub, aud and others. Public claims are described while using JWTs. IANA JSON web token registry is used for avoiding collisions. Private claims can be customized between ends based on mutual agreement. Signature is meant for secrecy which includes encoded header, encoded payload and algorithm. Structural representation of a sample JWT id is depicted in Fig.4.

Advantages of JWT

□ JSON is compact in size and can be used in various back ends and domains.

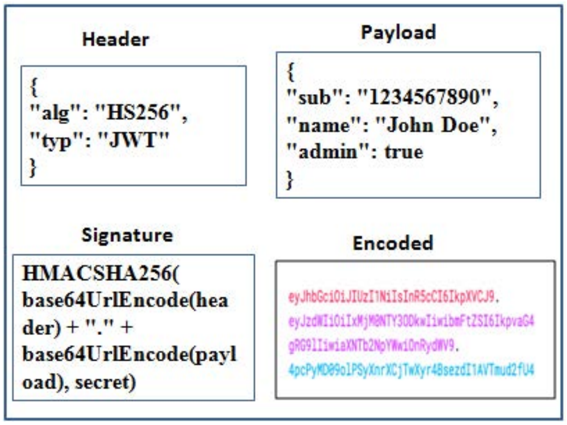
□ Cookies are not required and JWT can be stored in any storage options

□ Debugging is simple and provide improved performance

□ It is based on RFC7519 and supports many libraries such as .NET, Java, Ruby, Python, Javascript, PHP

□ Effective session management is enabled and it avoids use of session object o server

□ It is mobile friendly and applicable in android, iOS, Windows 8, etc.



```
{
"alg": "HS256",
"typ": "JWT"
}
```

```
{
"sub": "1234567890",
"name": "John Doe",
"admin": true
}
```

```
HMACSHA256(
base64UrlEncode(hea
der) + "." +
base64UrlEncode(payl
oad), secret)
```

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4
gRG9lIiwiaXNTb2NpYWwiOnRydWV9.
4pcPyMD09olPSyXnrXCjTwXyr4BsezdI1AVTmud2fU4
```

Fig. 4 Structure of sample JWT

IV. PROPOSED MODEL

Internet of Things (IoT) system is comprised with IoT devices with different software and hardware implementations. Further IoT devices are naturally Internet connection is implemented through Internet Protocol (IP) stack in IoT system. IP stack requires more power and memory because of its awfully multifaceted representation. Local connection of IoT devices is accomplished via non IP networks. Smart gateway is required for establishing internet connection in between devices. Here gateway acts as an interface amid internet and devices of IoT system. Utilization of intelligent gateways reduces the number of firewalls. Middleware support is required to meet the application requirements for the proposed model. IoT device registration, recognition and managing the databases are performed by middleware. In addition it ensures confidentiality and safety of data. REST API is helpful in providing verification as well as authorization. Then collected data is interpreted. Various authorization protocols are presented. OAuth is an example of authorization protocol in open access which permits resources to use username, password and tokens.

Fig.5 portrays the flow of secure data communication in the IoT system. IoT device, IoT gateway and servers (both application and authorization server) are key players in the IoT system. Connection model is based on requests and responses. Device send request for sending data to IoT gateway first. Then gateway ID and private key are forwarded to the servers in which authorization is ensured. Then server responds to gateway by sending JSON token. After that, request is sent by gateway for accessing JSON token. Once taken is received respective data is sent through authorization/application server. Further token verification is done with the help of GET query. Finally data with corresponding JSON token is sent to IoT gateway from device.

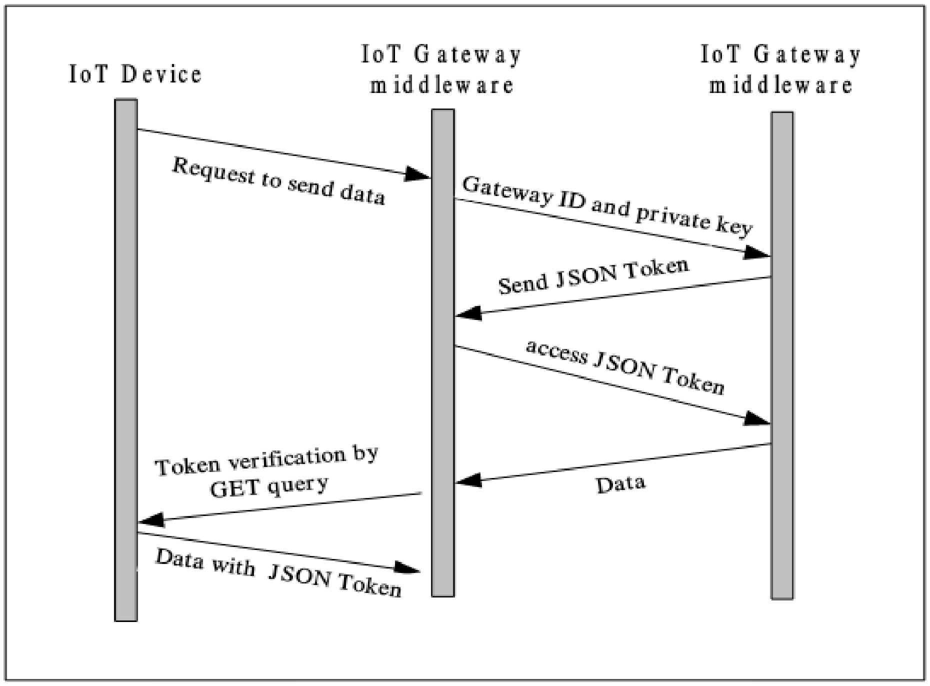


Fig.5 Data communication flow

Fig.6 depicts the architecture of proposed model and the flow of working is comprised in two steps. At the first step, authorization is applied for registering IoT devices on the basis of their legitimate account credentials. In step 2 gateways authenticate the requests coming from IoT devices. Payloads of the device are accessed via REST API. Then gateway will forward request to exposed API making use of identification parameters. Gateway ID and private key are considered as parameters. Further request is validated and authorized by API like gateway. Every method of REST API needs validation. If gateway is authorized then encrypted response is sent back via gateway. Finally after authentication, gateway allows to permission to devices which enables real time data traffic through gateway. The proposed model is developed o the basis of JSON web token and REST API.

IoT system requires effective data communication in quick and secure manner. API simplifies this and it is considered as potential for IoT. HTTP is utilized by RESTful web service in favor of M2M communication and to transfer files in JSON and XML. Distinct REST endpoints are described with respective of operations. Once authentication of gateway operations is completed, it is possible to use URLs. A jQuery client is built by gateway for pursuing RESTful Web Service. All requests of service are treated as URL. Service’s response includes the details device in JSON format.

While analyzing the proposed model and its associated various security features. Every part of the proposed system is examined to ensure secure data communication between the ends. The summary of analysis is listed as following.

□ Devices in the IoT system are connected to the gateway and it acts as an interface among them. Intelligent gateways are effective and provide protection against various attacks.

□ Secure data communication is carried making use of REST API.

□ Cryptographic algorithms are utilized for enhancing secure data communication.

□ Both verification and authorization are implemented by utilizing REST APIs in which complexity is low and easy accessible for all formats.

□ JSON web token ensures secrecy between the ends by implementing signature verification.

□ JWTs provide effective session man agent with the use of public or private pair of keys and cryptographic algorithms.

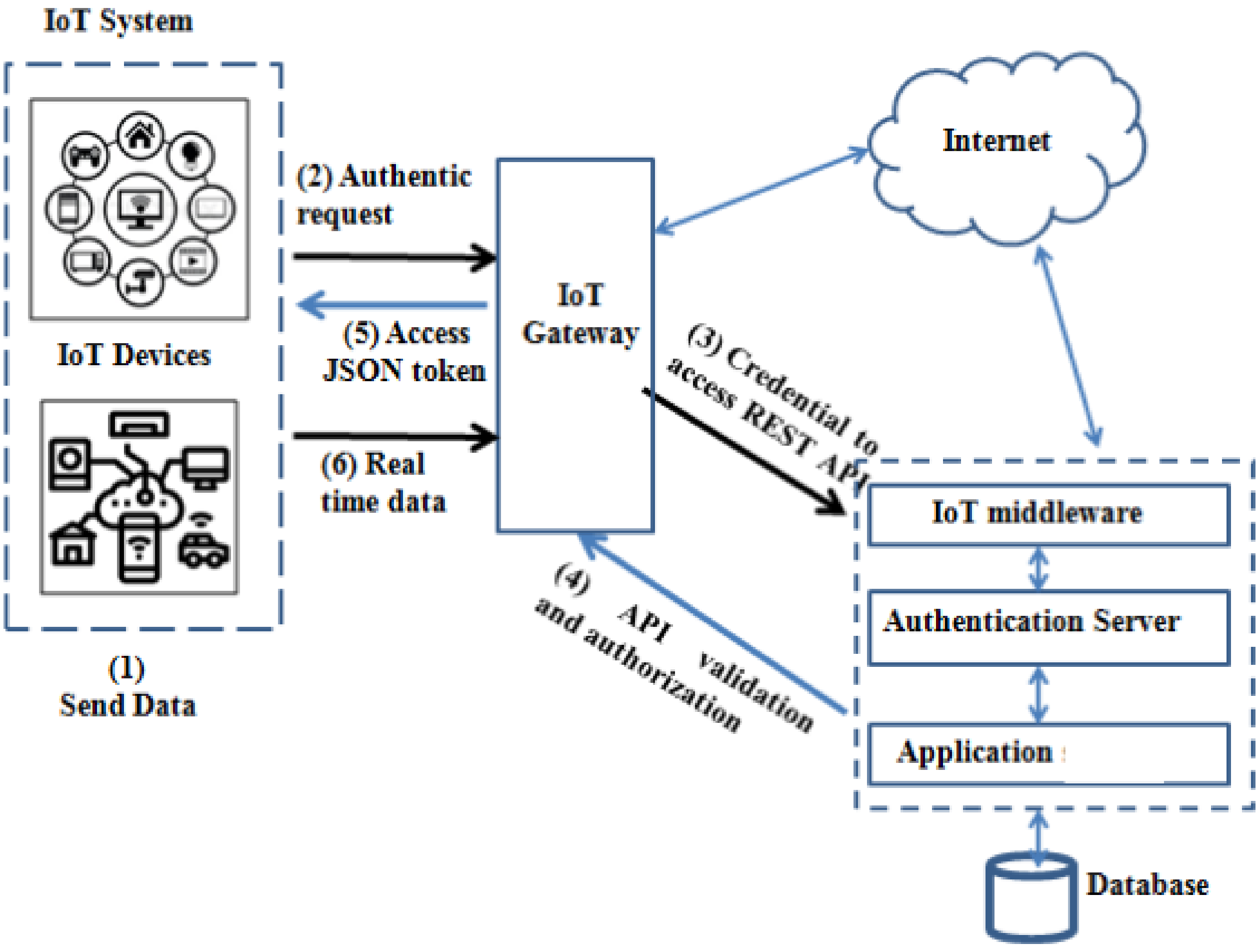


Fig.6 Architecture of proposed model

V. RESULTS AND DISCUSSION

The entire experimental setup is carried out using a secured standalone infrastructure hosted locally and in the cloud. A set of IoT nodes configured with Raspberry pi 4 with 8 GB RAM. These Raspberry pi nodes operate based on the local running web server implemented using REST and JSON web token to communicate with the social networking system called mini twitter for writing the data and updating the node user information. The entire experimentation is carried out with local and cloud runtime. Further, a CDN service (Cloudflare is used) is also enabled to distribute the content faster.

Table 1. Performance metrics - local runtime information

Application	SOAP	REST API	REST API + JWT
Total time in ms	58.26	57.46	53.12
Instances	2	2	2
Total time (%)	35%	35%	35%
Mount (ms)	58.26	57.46	53.12
Update in ms	48.55	48.55	45.55

Render in ms of 5MB standard file	66.25	66.12	64.12
Unmount (ms)	48.55	48.55	42.55
Reloading (ms)	48.55	48.55	42.55

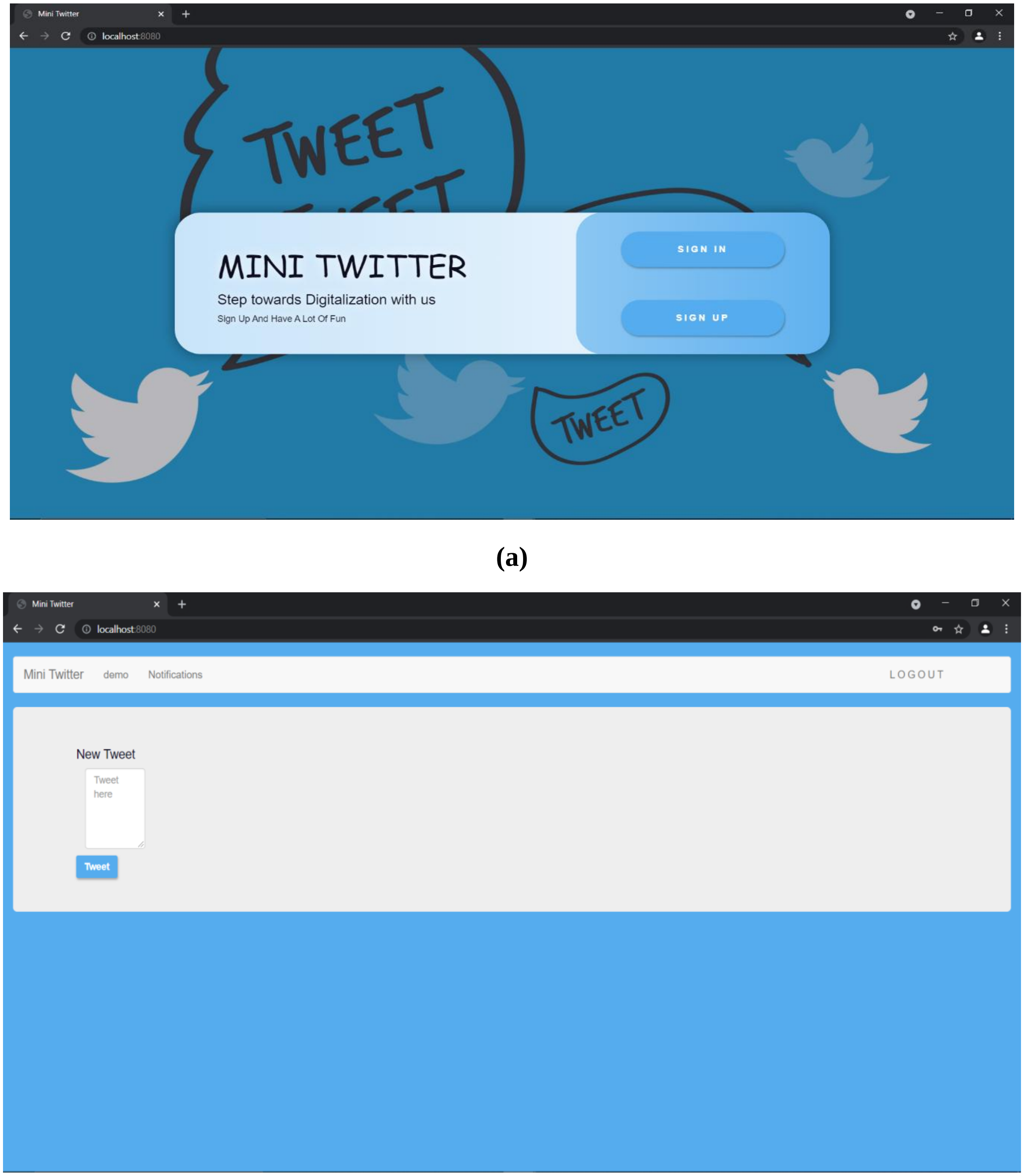
Table 2. Performance metrics - cloud runtime information

Application	SOAP	REST API	REST API + JWT
Total time in ms	68.26	67.46	58
Instances	1	1	1
Total time (%)	45%	45%	45%
Mount (ms)	48.26	47.46	43.12
Update in ms	58.55	58	55
Render in ms of 5MB standard file	72	72	64
Unmount (ms)	55	55	55
Reloading (ms)	52	52	47

Table 3. Performance metrics - cloud runtime information with Cloudflare

Application	SOAP	REST API	REST API + JWT
Total time in ms	48	47	45
Instances	1	1	1
Total time (%)	45%	45%	45%
Mount (ms)	35	35	32
Update in ms	40	40	35
Render in ms of 5MB standard file	40	40	38
Unmount (ms)	35	35	35
Reloading (ms)	28	28	21

Table 1 shows the performance metric analysis results of local runtime information. Table 2 shows the performance metric analysis results of cloud runtime information. Table 3 shows the performance metric analysis with CDN deployment. From the experimental analysis, it is observed that the CDN deployment enhances the overall performance of data communication. Fig.7 shows the screenshot of the social networking system web application.



(a)

(b)

Fig.7 Social networking system

VI. CONCLUSION

In this research paper secure end to end encryption enabled communication systems is deployed for Internet of Things (IoT)supported social networking system. Secure data communication is aimed by integrating the security features of REST API and JSON web token. IoT device in the system creates request for establishing connection and sharing data through IoT gateway. Authentication and authorization are also considered with this proposed research work. JSON web token is useful for signature verification which defines integrity of data and its communication. IoT gateway acts as intermediate entity amid IoT system and servers. Real time data traffic is secured by utilizing both REST API and JSON web token.

REFERENCES

1. Cisco. The Internet of Things Reference Model. Accessed: Aug. 23, 2020. [Online]. Available: http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf
2. Curtin University. Introduction to the Internet of Things. Accessed: Aug. 23, 2020. [Online]. Available: https://study.curtin.edu.au/offering/mooc-introduction-to-the-internet-of-things_iot1x
3. V. Karagiannis, P. Chatzimisios, F.Vazquez-Gallego, and J. Alonso-Zarate, ``A survey on application layer protocols for the Internet of Things," Trans. IoT Cloud Comput., vol. 3, no. 1, pp. 11_17, 2015.
4. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, ``Internet of Things (IoT): A vision, architectural elements, and future directions," Future Gener. Comput. Syst., vol. 29, no. 7, pp. 1645_1660, 2013, doi: 10.1016/ j.future.2013.01.010.
5. Statista. IoT: Number of Connected Devices Worldwide 2012_2025. Accessed: Aug. 23, 2020. [Online]. Available: statistics/471264/iot-number-of-connected-devices-worldwide
6. O. Ethelbert, F. F. Moghaddam, P. Wieder and R. Yahyapour, "A JSON Token-Based Authentication and Access Management Schema for Cloud SaaS Applications," 2017 IEEE 5th International Conference on Future Internet of Things and Cloud (FiCloud), Prague, 2017, pp. 47-53. doi: 10.1109/FiCloud.2017.29
7. O. Ethelbert, F. F. Moghaddam, P. Wieder and R. Yahyapour, "A JSON Token-Based Authentication and Access Management Schema for Cloud SaaS Applications," 2017 IEEE 5th International Conference on Future Internet of Things and Cloud (FiCloud), Prague, 2017, pp. 47-53. doi: 10.1109/FiCloud.2017.29
8. R. Buyya, J. Broberg, and A. M. Goscinski, Cloud Computing: Principles and Paradigms. Hoboken, NJ, USA: Wiley, 2011.

9. Y. Ai, M. Peng, and K. Zhang, ``Edge cloud computing technologies for Internet of Things: A primer," Digit. Commun. Netw., vol. 4, no. 2,pp. 77_86, 2018.

10. https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet

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

12. 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.

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

14. S. Bandyopadhyay and A. Bhattacharyya, ``Lightweight Internet protocols for Web enablement of sensors using constrained gateway devices," in Proc. Int. Conf. Comput., Netw. Commun. (ICNC), Jan. 2013, pp. 334_340, doi: 10.1109/iccnc.2013.6504105.

15. P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttila and M. Sain, "Lightweight and Secure Session-Key Establishment Scheme in Smart Home Environments," in IEEE Sensors Journal, vol. 16, no. 1, pp. 254-264, Jan.1, 2016. doi: 10.1109/JSEN.2015.2475298.

16. K. Gutzmann, "Access control and session management in the HTTP environment," in IEEE Internet Computing, vol. 5, no. 1, pp. 26-35, Jan.-Feb. 2001. doi: 10.1109/4236.895139

17. G. Cassar, P. Barnaghi, W. Wang, and K. Moessner, “A hybrid semantic matchmaker for IoT services,” in Proceedings – 2012 IEEE Int. Conf. on Green Computing and Communications, GreenCom 2012, Conf. on Internet of Things, iThings 2012 and Conf. on Cyber, Physical and Social Computing, CPSCom 2012, 2012, pp. 210–216.

18. J. Boman, J. Taylor, and A. Ngu, “Flexible IoT Middleware for Integration of Things and Applications,” Proc. 10th IEEE Int. Conf. Collab. Comput. Networking, Appl. Work., no. CollaborateCom, pp. 481–488, 2014.

19. Wu, F., Xu, L., Kumari, S., & Li, X. (2017). “A privacy-preserving and provable user authentication scheme for wireless sensor networksbased on internet of things security”. Journal of Ambient Intelligence and Humanized Computing, 8(1), 101-116.

20. Y. Zhang, L. Duan, and J. L. Chen, “Event-driven SOA for IoT services,” in Proceedings - 2014 IEEE International Conference on Services Computing, SCC 2014, 2014, pp. 629–636.

21. 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.

22. M. de Souza Lima, A. de Ribamar L. Riberio, and E. David Moreno, “Proposal of a Standard Vocabulary for Services Discovery on the Internet of Things,” Proc. 11th Int. Conf. Web Inf. Syst. Technol., pp. 129–134, 2015.

23. K. Hur, S. Chun, X. Jin, and K. H. Lee, “Towards a Semantic Model for Automated Deployment of IoT Services across Platforms,” in Proceedings - 2015 IEEE World Congress on Services, SERVICES 2015, 2015, pp. 17–20.

24. K. Hur, X. Jin, and K. H. Lee, “Automated deployment of IoT services based on semantic description,” in IEEE World Forum on Internet of Things, WF-IoT 2015 - Proceedings, 2016, pp. 40–45.

25. A. H. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and M. Z. Sheng, “IoT Middleware: A Survey on Issues and Enabling technologies,” IEEE Internet Things J., vol. X, no. X, pp. 1–1, 2016.

26. M. Nordahl and B. Magnusson, “A lightweight data interchange format for internet of things with applications in the PalCom middleware framework,” J. Ambient Intell. Humaniz. Comput., vol. 7, no. 4, pp. 523–532, 2016.

27. RFC5988, Available online 28.3.2017 at https://tools.ietf.org/html/rfc5988

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

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

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

31. H. Garg and M. Dave, "Securing IoT Devices and SecurelyConnecting the Dots Using REST API and Middleware," 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), 2019, pp. 1-6, doi: 10.1109/IoT-SIU.2019.8777334.

32. S. Ahmed and Q. Mahmood, "An authentication based scheme for applications using JSON web token," 2019 22nd International Multitopic Conference (INMIC), 2019, pp. 1-6, doi: 10.1109/INMIC48123.2019.9022766.

33. M. Jones, J. Bradley, and N. Sakimura, “JSON Web Token (JWT) RFC 7519”, http://www.rfc-edi tor.org/rfc/rfc7519.txt, RFC Editor 2015.

34. L. Viktor Jánoky, J. Levendovszky, P. Ekler, "An analysis on the revoking mechanisms for JSON Web Tokens," International Journal of Distributed Sensor Networks, vol. 14, September 2018, doi: https://doi.org/10.1177/1550147718801535

35. N. Hong, M.Kim, M. Jun, J. Kang, "A Study on a JWT-Based User Authentication and API Assessment Scheme Using IMEI in a Smart Home Environment," in jornal of sustainability, vol. 9, no. 7, June 2017

36. E. O. Zaballa, C. Vanhelleputte, S. Ruepp, M. N. Petersen and M. S. Berger, "Design and implementation of a cloud-based multi-LPWAN IoT data collection service," 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), 2020, pp. 1-2, doi: 10.1109/WF-IoT48130.2020.9221200.

37. https://www.redhat.com/en/topics/api/what-is-a-rest-api

38. https://jwt.io/introduction