

Access mechanism using inter planetary file system

Pandu Ranga Reddy Konala ^{1*}, V. SaiAbhinav Reddy ¹, D. Krishna Chaitanya Varma ¹, Radhika. G ¹

¹ Dept. of Computer Science and Engineering, Amrita School of Engineering, Coimbatore. India

*Corresponding author E-mail: CB.EN.U4CSE15422@cb.students.amrita.edu

Abstract

With the advent of Internet of Things, trust is one of the important factor for provisioning secure, reliable, seamless communications and services. Connected devices are the order of the day, where IoT plays a vital role in the functioning. However, these IoT devices use various protocols for communication mostly a centralized architecture for data transfer. Some of these protocols are Message Queue Telemetry Transport (MQTT), which is a Pub-Sub (publisher and subscriber) protocol used for a device-to-device communication on TCP through a broker. This centralized server can be a third party service provider, which again raises the issue of trust. To address this, we propose a decentralized and distributed network architecture called Inter Planetary File System (IPFS) is used instead of third-party service provider. IPFS enables publish – subscribe model similar to MQTT, but by removing the broker and making a peer-to-peer communication. Peer-to-peer architecture has its own challenges like security. The issue of security is also addressed by encrypting the files or data that has to be sent over the IPFS network using PGP (Pretty Good Privacy).

Keywords: Interplanetary File System; Message Queue Telemetry Transport; Internet of Things; Peer-to-Peer Communication.

1. Introduction

The Connected devices also called as Internet of Things (IoT) is a global industry movement which brings together people, process, data and things are more relevant and valuable than ever before. The innovative developments in the area of digital technology along with the evolution of internet have paved way for the use of IoT in various applications from the past decade [1]. This technology can be used to solve diversified problems in various sectors like automobile, medical and other allied areas. Though IoT has a good potential in the digital world, it comes across several issues during its deployment, with respect to build quality of devices, identity, management and mainly secure communication between devices [2].

Security also plays a major role in communication between devices. There are many protocols that IoT devices use in which most of them are centralized which have intermediate node (broker) for communication. In the recent days many cloud computing services emerged, which have quickly become popular for Big Data generated by numerous IoT devices for processing [3]. Among the Cloud Service Providers, most of them offer the Data Owners with flexible approaches to store the datasets remotely in the cloud and facilitates the owners to change/update these datasets. Most of the IoT applications, which choose to store and process data on the cloud. Providing security will be the main concern, as Cloud Storage Service (CSS) is not secure by nature. "Cloud users will not have no control over the cloud storage servers being used, which means there are risks of Data Confidentiality, Data Integrity, and Data Availability" [4]. MQTT is an open pub/sub protocol [5] designed for constrained devices used in telemetry applications [6]. MQTT is designed in such away that its client's side implementation is made very simple. All of the system complexities reside on the broker's side. According to [6] the routing or networking techniques are not specified by MQTT, but instead the assumption made by MQTT is that the underlying network provides a "point-

to-point, session-oriented, auto-segmenting data transport service with in-order delivery (such as TCP/IP)" [6]. As mentioned earlier, this intermediate node can be a cloud service provider or any third-party database service provider. As the data is being stored in these third party service providers, there are risks that the client should take like lack of trust and loss of control.

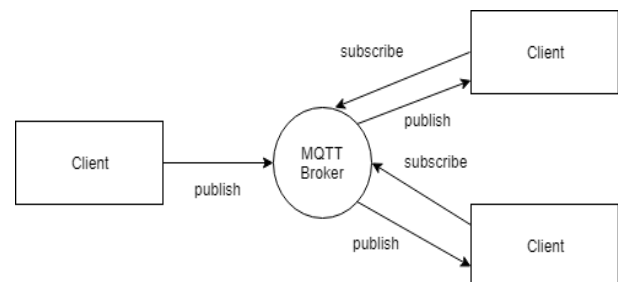


Fig. 1: MQTT Publish Subscribe Using Broker as an intermediate Node.

Considering the above discussed problems, a decentralized and distributed architectural based protocol like IPFS can be used to address the issue of broker. IPFS is a protocol designed to create a peer-to-peer scheme of storing and sharing data repositories of various kinds in a distributed file system. It provides high performance and clustered persistence. While delivering large amount of data to users, a peer-to-peer approach could save millions of bandwidths. High latency networks are a real barrier of entry to developing world. IPFS provides resilient access to data, independent of low latency or connectivity to the backbone. It has a publish-subscribe model which focuses on reliability, delivery guarantee and data persistency data persistency.

2. Literature survey

Many researches and authors have long been using MQTT for IoT and most of them stressed on privacy and security. In [7], ensuring privacy for IoT was given higher priority by applying ABE, grounded on nonspecific Pub-Sub architecture. In this technique a payload is used to encrypt by using Advance Encryption Standard (AES) algorithm which comes under Symmetric key cryptography is used to encrypt the payload and to make sure that the payload size and cipher text size as same, AES key is encrypted with the help of ABE. In [8] the authors argue that, both these encryption techniques i.e. AES and ABE are used to accomplish encryption on limited bits of data which are generated by the IoT will be a computational overhead for IOT devices. Hence in [8] the authors aimed at the optimization of ABE's complex arithmetic operations using suitable cryptography parameters (MQTT-S) instead of performing double encryptions. In [9] the authors proposed a middleware based on Pub-Sub architecture in which CP-ABE and Predicate based encryption is enabled to protect the privacy of subscriber's interest and the published content's confidentiality. In [10] the authors present the strategy and execution of IoT based Home Automation, a steadfast WSN technology will be interconnected through MQTT (Telemetry Transport) protocol to establish the communication among diverse devices via Ngrok which is a third-party cloud access provider where the author completely trusts their services, which might lead to a point of failure and does not unravel the speed of light problem.

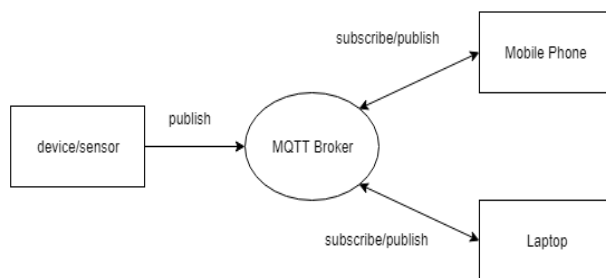


Fig. 2: MQTT Publish Subscribe Model.

The challenging aspect in IoT is to introduce privacy for users' data. So, it is necessary to develop a decentralized IoT, which should be designed to have a built-in privacy [11]. With the decentralized IoT data management, users will have a choice to share and sell the sensor data to third-party entities without any need of intermediaries.

According to the authors [11] the objective therefore, is to ensure that the user data is not delivered to the centralized entities by providing a decentralized data access model for IoT. To realize this goal the Blockchain techniques and peer-to-peer communication will play a vital role [12], [11]. The InterPlanetary File System, "IPFS", [13] is a cutting-edge peer-to-peer distributed file system that seeks to connect all computing devices with the similar system of files i.e., by trading objects with each other. Unlike all other Telemetry Transport protocols, IPFS also has a publish-subscribe feature which does not require a broker for data transmission. Which makes it more resilient to several network-based criticisms. IPFS has no single point of failure, and nodes do not need to trust each other which is a perfect alternative for the MQTT protocol in our research work.

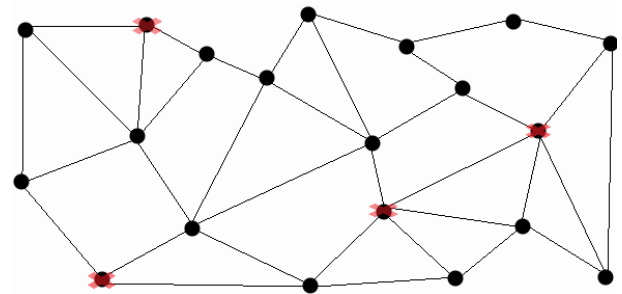


Fig. 3: Distributed Network.

As previously discussed, centralized IoT devices will have scalability issues in regard to data management and access control which force the users to give their entire data to the third-party brokers or intermediaries for managing their data and thus losing the data privacy [14]. This problem has led to the research on working with the blockchain technology with the help of peer-to-peer data storage mechanisms [11]. Bitcoin's success is the main motivation behind the research on using the blockchain and peer-to-peer technology. In [15] the authors have stressed on the capability of blockchains in maintaining the data exchanges through an immutable log as well as performing the access control. Access policies around the public key infrastructure of blockchain networks have been created from where the access control element comes from [16].

IBM Adept [17] [11] is a combined effort from IBM and Samsung whose aim is to connect the blockchain and to develop a decentralized platform for IoT. For peer-to-peer communication, Adept uses TeleHash, and for peer-to-peer file sharing, an Ethereum blockchain development platform on top of BitTorrent is used. According to [11] "the issues IBM Adept faces in implementing a blockchain-based solution for decentralizing IoT are the poor scalability of blockchains and the inherent latency in blockchain consensus". The authors in [15] suggest dividing the IoT blockchain network into smaller sub-networks, since a single blockchain cannot take the load [18]. The author has explored about various aspects of the file system i.e., which implies security, transparency and data privacy of instigating along with The Internet of Things for a better data and healthy algorithm opacity which lacks the internal protocol security i.e., Encryption. This problem can be solved by open source tool OpenPGP. Thus, capturing all these characteristics into contemplation, a typical order of stages for our problem statement is: Initialize the IPFS Daemon-ipfsinit(), Setting the Topic for publish-subscribe, Add Listeners, Encrypt Channel data by Asymmetric Key Encryption, Run scenarios using various cases (Related to accessibility). Impact of Spoofing on IoT devices are one of the major problems and can be secured by cryptographic algorithms [20].

3. Background

3.1. IPFS

The InterPlanetary File System (IPFS) is a distributed file storage protocol that allows computers all over the world to store and serve files as part of a giant, peer-to-peer network. It is also called as Distributed Web. Every single computer that is running IPFS acts as both a client and a server. In other words, each computer running the IPFS software can serve content to any other computer in the network, as well as request content from anyone in the network. Every file added to IPFS is given a unique address derived from a hash of the file's content. IPFS addresses are multi-hashes, which combine information about the hashing algorithm used as well as the hash output into a single string.

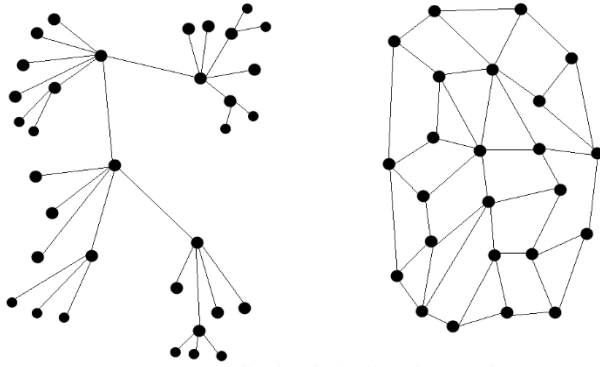


Fig. 5: Decentralized and Distributed Networks.

IPFS uses Merkle data format for hashing. Every Merkle tree is a directed acyclic graph (DAG) where each node is accessed by its name. In Merkle each branch represents the hash of its local contents and naming the children by their hash instead of their full contents. This restricts to edit a node once it is created by preventing cycles, there is no provision to link the first created node to the last node as the reference.

IPFS relies on a distributed hash table (DHT), i.e., mapping from hash to some people who may have the content addressed by that hash. The hash table is distributed because no single node in the network holds the whole thing. Instead, each node stores a subset of the hashtable, as well as information about which nodes are storing other relevant sections.

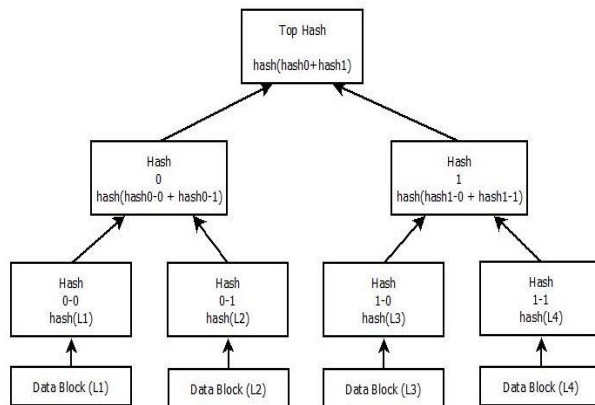


Fig. 6: Merkle DAG Hashing Method.

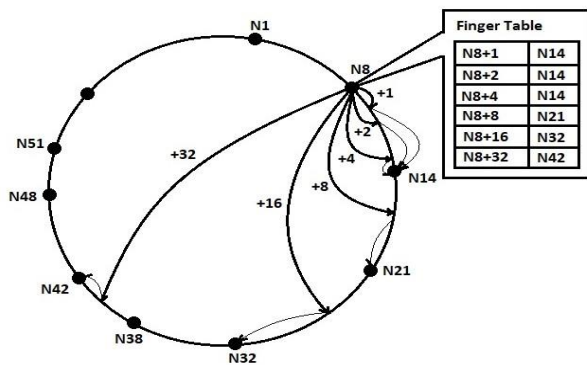


Fig. 7: Distributed Hash Table.

3.2. PUB-SUB model

Publish-Subscribe, called 'pubsub' for short, is a pattern often used to handle events in large-scale networks. 'Publishers' send messages classified by topic or content and 'Subscribers' receive only the messages from the 'Topics they are interested in, all without direct connections between publisher and subscribers. This approach offers much greater network scalability and flexibility.

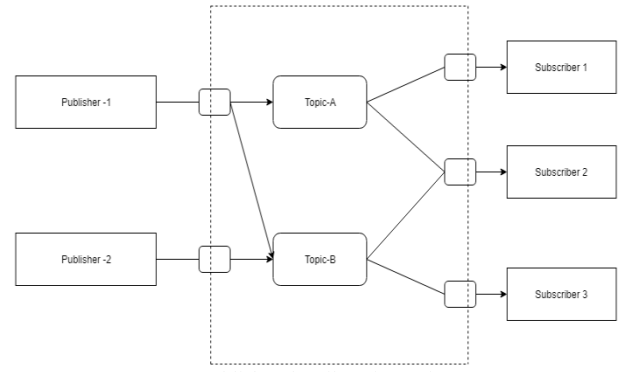


Fig. 8: Publish Subscribe Via Topic MoC.

3.3. Pretty good privacy

Pretty Good Privacy (PGP) provides a combination of cryptographic methodologies such as digital signatures, hashing and public-key cryptography to keep data protected. Each public key is bound to a username or an electronic mail address. This procedure can be used to encrypt text files, emails, data files, directories and disk partitions. OpenPGP is a standard of PGP that is open-source for public use. The GPG software is an independent implementation of the OpenPGP standards, so one can use it to exchange encrypted messages with people using other OpenPGP implementations.

PGP can be used to send messages confidentially by combining symmetric-key encryption and public-key encryption. The message is encrypted using asymmetric encryption algorithm, which requires asymmetric key. Each symmetric key is used only once and is also called a session key. The message and its session key are sent to the receiver. The session key must be sent to the receiver so they know how to decrypt the message, but to protect it during transmission it is encrypted with the receiver's public key. Only the private key belonging to the receiver can decrypt the session key.

It also supports message authentication and integrity checking. Because the content is encrypted, any changes in the message will result in failure of the decryption with the appropriate key. The sender uses PGP to create a digital signature for the message with either the RSA or DSA algorithms. To do so, PGP computes a hash from the plaintext and then creates the digital signature from that hash using the sender's private key.

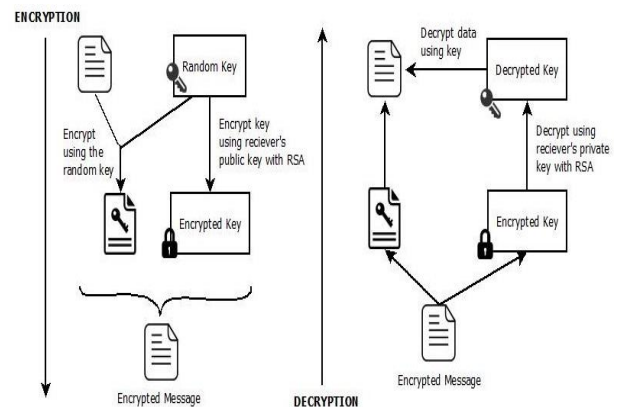


Fig. 9: PGP Encryption and Decryption.

4. Proposed architecture

4.1. Stage 1: (publish - subscribe)

Enabling Publish Subscribe between the end user (PC) and IoT device (Raspberry Pi2). When a message is published from the PC, it will be received by both the PC and the Pi as PC is self-subscribed to the topic and Pi is also subscribed to that topic.

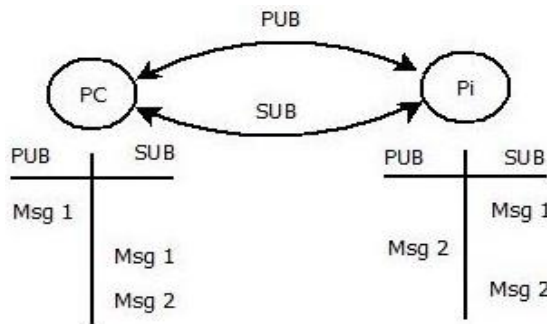


Fig. 10: Pub-Sub between PC and Pi.

4.2. Stage 2: (transfer files over IPFS)

Adding files to IPFS network and getting the hash for the uploaded file and publishing it to the IPFS network through pub-sub model.

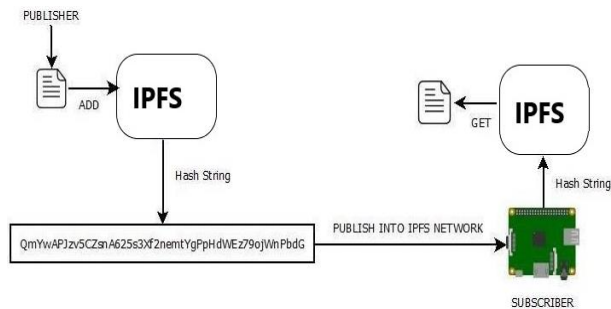


Fig. 11: File Transfer through IPFS Network.

4.3. Stage 3: (encryption using PGP)

While sending Hash through the IPFS network encryption is being done for allowing only the authorized users to decrypt the file and access the contents of it. PGP (Pretty Good Privacy) is used to encrypt the file using symmetric-key encryption and public-key encryption.

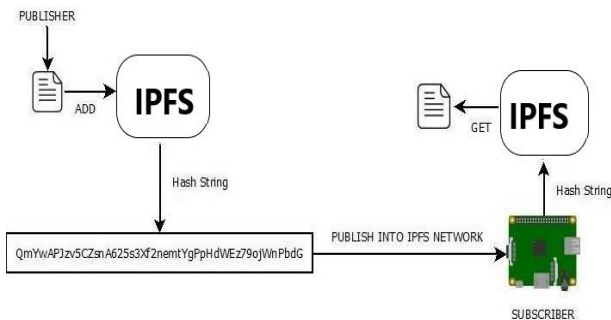


Fig. 12: File Encryption Using PGP.

4.4. Stage 4

Changing the state of servo based on the data from the decrypted file at the subscriber.

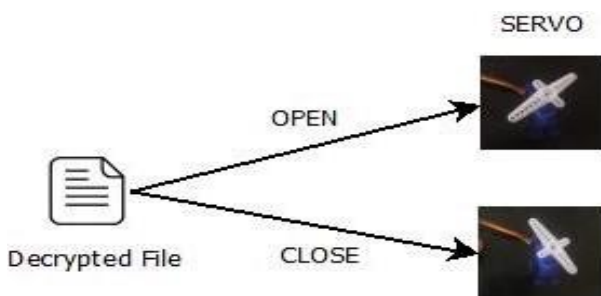


Fig. 13: Setting Servo State from File.

5. Results

5.1. Daemon running through console

Starting the IPFS network by running the following command "ipfs daemon --enable-pubsub-experiment"

```

bob@ubuntu:~$ ipfs daemon --enable-pubsub-experiment
Initializing daemon...
Successfully raised file descriptor limit to 2048.
Swarm listening on /ip4/127.0.0.1/tcp/4001
Swarm listening on /ip4/192.168.146.173/tcp/4001
Swarm listening on /ip6:::1/tcp/4001
Swarm announcing /ip2p-circuit/ipfs/QmZde4FzsuZVcLLdQ5VQeX8T6SpRS00vZRBovsgqGJF0H5
Swarm announcing /ip4/127.0.0.1/tcp/4001
Swarm announcing /ip4/192.168.146.173/tcp/4001
Swarm announcing /ip6:::1/tcp/4001
API server listening on /ip4/127.0.0.1/tcp/5001
Gateway (readonly) server listening on /ip4/127.0.0.1/tcp/8080
Daemon is ready
  
```

Fig. 14: Daemon Running in Console.

5.2. Daemon running through the java applet

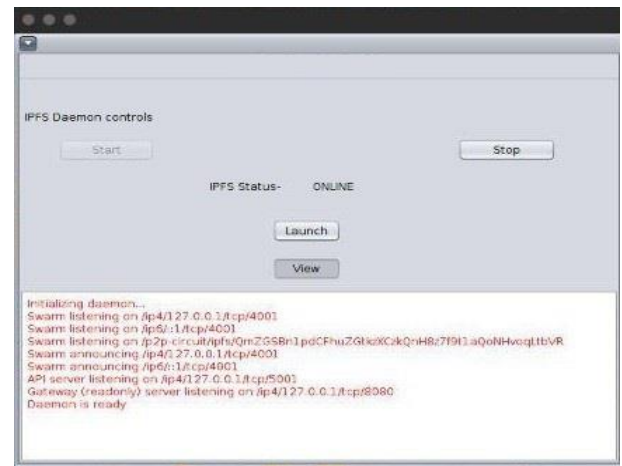


Fig. 15: Daemon Running Through Java Applet.

5.3. Controlling the servo using java applet

Servo can be controlled by clicking on "OPEN" and "CLOSE" buttons

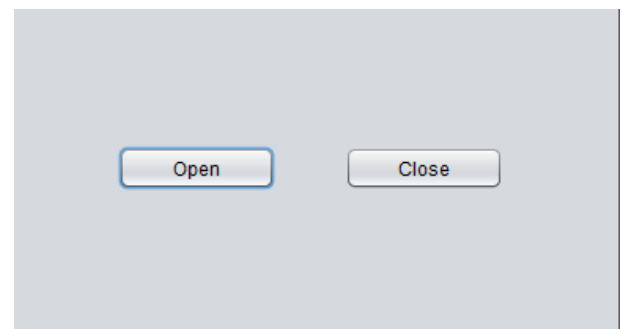


Fig. 16: Controls for Servo through Java Applet.

5.4. Servo results in terminal

Checking the published value in terminal by running the following command "ipfssubsub sub servo" 0 – Close AND 1 – Open

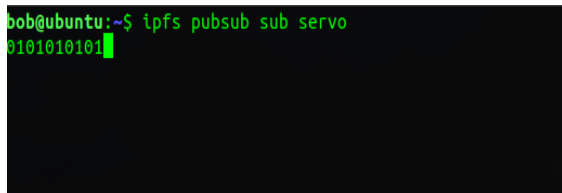


Fig. 17: Servo Results in Terminal.

5.5. Servo results in debugger console

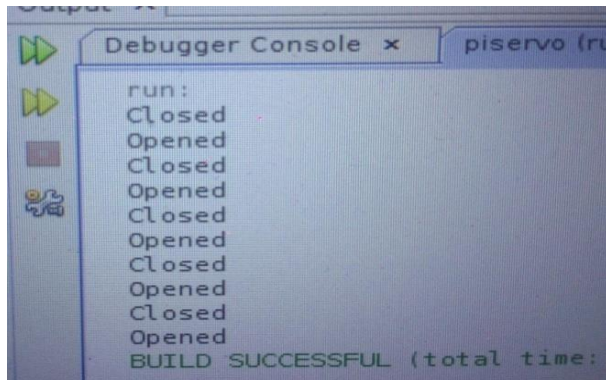


Fig. 18: Servo Results in Debugger Console.

5.6. Servo states

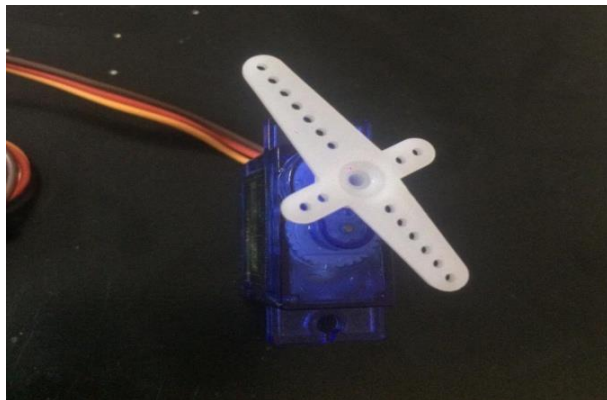


Fig. 19: Servo in Open State.

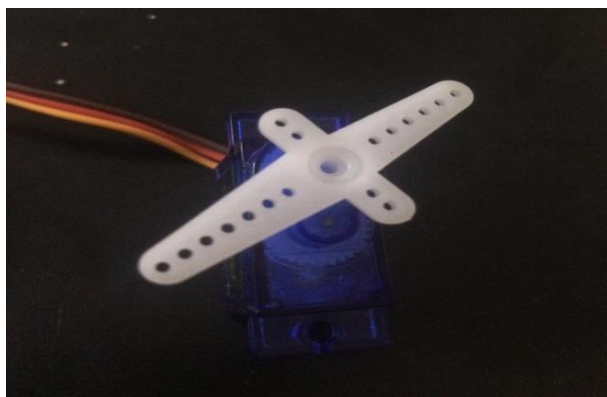


Fig. 20: Servo in Closed State.

6. Conclusion

Inter Planetary File System indeed plays a major role in the field of D2D communication. It is not only keeps the data secure but also guarantees a total control of the device which can be accessible from any part of the world due to its distributed architecture. The paper presents the use of IPFS in the field of Internet of things as a communication protocol which solves the problem of data accessibility and availability by scaling out centralized servers and

cloud platforms. Which makes it reliable and inexpensive for users unlike MQTT.

References

- [1] B. S. Adiga, P. Balamuralidhar, M. A. Rajan, R. Shastry, and V. L. Shivraj, "An Identity Based Encryption Using Elliptic Curve Cryptography for Secure M2M Communication," in Proceedings of the First International Conference on Security of Internet of Things, ser. SecurIT '12. ACM, 2012, pp. 68–74.
- [2] D. Díaz Pardo de Vera, A. Siguñenzalquiero, J. Bernat Vercher, and L. A. Hernández Gómez, "A Ubiquitous sensor network platform for integrating smart devices into the semantic sensor web," vol. 14, no. 6. Multidisciplinary Digital Publishing Institute, 2014, pp. 10 725–10 752.
- [3] H. S. Narman, M. S. Hossain, M. Atiquzzaman, and H. Shen, "Scheduling internet of things applications in cloud computing," *Annales des Télécommunications*, 72(1-2):79–93, 2017.
- [4] Liu, Bin, et al. "Blockchain based data integrity service framework for IoT data." Web Services (ICWS), 2017 IEEE International Conference on IEEE, 2017. <https://doi.org/10.1109/ICWS.2017.54>.
- [5] "MQ Telemetry Transport," <http://mqtt.org>.
- [6] Hunkeler, Urs, Hong Linh Truong, and Andy Stanford-Clark. "MQTT-S—A publish/subscribe protocol for Wireless Sensor Networks." Communication systems software and middleware and workshops, 2008. Comsware2008. 3rd international conference on. IEEE, 2008.
- [7] X. Wang, J. Zhang, E. Schooler, and M. Ion, "Performance evaluation of Attribute-Based Encryption: Toward data privacy in the IoT," in Communications (ICC), 2014 IEEE International Conference on, June 2014, pp. 725–730.
- [8] P. Pal, G. Lauer, J. Khoury, N. Hoff, and J. Loyall, "P3S: A Privacy Preserving Publish-subscribe Middleware," in Proceedings of the 13th International Middleware Conference, ser. Middleware '12, pp. 476–495. https://doi.org/10.1007/978-3-642-35170-9_24.
- [9] "ZigBee Alliance," <http://www.zigbee.org>.
- [10] Agarwal, A., Singh, R., Gehlot, A., Gupta, G., Choudhary, M.: IoT enabled home automation through nodered and MQTT. *Int. J. Control Theor. Appl.* (2017). ISSN 0974-5572
- [11] Ali, Muhammad Salek, Koustabh Dolui, and Fabio Antonelli. "IoT data privacy via blockchains and IPFS." Proceedings of the Seventh International Conference on the Internet of Things. ACM, 2017.
- [12] Marco Conoscenti, Antonio Vetro and Juan C. D. Martin. "Blockchain for the Internet of Things: A Systematic Literature Review." In Proceeding of The Third International Symposium on Internet of Things: Systems, Management and Security (IOTSMS-2016). <https://doi.org/10.1109/AICCSA.2016.7945805>.
- [13] Benet, Juan. "IPFS-content addressed, versioned, P2P file system." arXiv preprint arXiv:1407.3561 (2014).
- [14] Ouaddah Aafaf, Anas Abou Elkalam, and Abdellah Ait Ouahman. "Towards a Novel Privacy-Preserving Access Control Model Based on Blockchain Technology in IoT." Europe and MENA Cooperation Advances in Information and Communication Technologies. Springer International Publishing, 2017.
- [15] Zyskind, Guy, Oz Nathan, and Alex Pentland. "Enigma: Decentralized computation platform with guaranteed privacy." arXiv:1506.03471 (2015).
- [16] Loise Axon. 2015. Privacy-awareness in Blockchain-based PKI. Retrieved April 12, 2017 from <http://goo.gl/3Nv2oK>
- [17] Device Democracy: Saving the Future of the Internet of Things. Retrieved May 10, 2017 from <https://goo.gl/18Y16F>
- [18] Huckle, Steve, et al., "Internet of things, blockchain and shared economy applications," *Procedia computer science* 98(2016): 461-466. <https://doi.org/10.1016/j.procs.2016.09.074>.
- [19] Arvind P Jayan, Harini N, A Scheme to Enhance the Security of MQTT Protocol, *International Journal of Pure and Applied Mathematics*, Volume 119 No. 12 2018, 13975-13982.