ISSN 2277-2685

IJESR/June. 2025/ Vol-15/Issue-3s/110-121

Jajeemogala Durga *et. al.*, / International Journal of Engineering & Science Research

A Secure And Cost-Efficient Blockchain Facilitated Iot Software Update Framework

N. Sudha Laxmaiah, G. Neha Latha, J. Meghana Sri

[1]Assistant Professor, Department Of Cse, Bhoj Reddy Engineering College For Women, India.

[2,3]B. Tech Students, Department Of Cse, Bhoj Reddy Engineering College For Women, India.

ABSTRACT

As resource-constrained Internet-of-Things (IoT) devices become popular targets of various malicious attacks, frequent updates to keep their software up to date are essential to their security. However, state-of-the-art software delivery and payment systems incorporate multiple services in a client-server structure requiring multiple transits of information between client and server, while also creating a wide attack surface. We propose a blockchain-based end-to-end secure software update delivery framework for Internet of Things (IoT) devices, which aims to ensure confidentiality, integrity, availability, efficiency, and audit-ability for verified software delivery, while offloading the cryptographic computation from resourceconstrained IoT devices to a decentralized blockchain system. In particular, we leverage Ciphertext-Policy Attribute-Based Encryption (CP-ABE) and design a customized authorization policy to not only ensure that software updates can only be decrypted and installed on authorized IoT devices but also significantly reduce the computational overhead for key generation and key delivery on the manufacturer side. Furthermore, secure and atomic software delivery and payments between IoT devices and the manufacturer are assured through smart contracts. The authenticity of the delivered software is guaranteed by offloading the computation-based signature validation to smart contracts. Compliance audits are satisfied through immutable records on the blockchain's public ledger, and the smart contracts efficiently guarantee the delivery of software updates in exchange for payment. Security analysis and experiments are performed to compare the proposed framework with state-of-the-art studies and validate its effectiveness.

1-INTRODUCTION

Internet of Things (IoT) devices are devices with sensors and processors that communicate over the Internet to perform specific tasks [1], [2]. Common examples of IoT devices include wearable medical devices that monitor a patient, smart home devices that control home systems, sensors in farming that report temperature and weather conditions, and many more. Due to broad consumer acceptance, in 2020, 749 billion USD was spent worldwide on IoT devices. Spending will surge to over 1,100 billion USD in 2023 [3], [4], [5]. There were 22 billion connected IoT devices in 2018. The number of IoT devices will double to over 50 billion by 2030 [5], [6]. However, due to their prevalent adoptions, while lacking sufficient computing resources for sophisticated security mechanisms, IoT devices can be easily compromised. One of the most well-known attacks is the Mirai botnet attack of 2016 which brought down large portions of the Internet through a DDoS attack waged by IoT devices [7], [8], [9], [10]. One essential protection approach to preventing such attacks is to patch the software of IoT devices frequently to ensure they are up to date. However, malicious attackers can also launch attacks against the software update process itself by, for example, providing manipulated software updates, or

retrieving software updates without payment. An example is an attack in 2016 that targeted the electric power grid in Ukraine disabling power for 30 substations. The outage impacted approximately 230,000 residents by updating firmware on IoT devices that controlled power systems with malicious software [11], [12]. There have also been software repository state attacks against package managers [13] that provide software libraries and components. Therefore, ensuring the confidentiality, integrity, and availability of the software update process is critical.

Problem statement:

The primary goal of this project is to establish a robust and secure framework for updating IoT (Internet of Things) devices across various domains. IoT devices are used in various fields, such as patient body temperature monitoring and traffic monitoring, and ensuring their software remains up-to-date is essential for their proper functioning and security.

2-FEASIBILITY STUDY

Feasibility Study

A feasibility study evaluates a project's or system's practicality. As part of a feasibility study, the objective and rational analysis of a potential business or venture is conducted to determine its strengths and weaknesses, potential opportunities and threats, resources required to carry out, and ultimate success prospects. Two criteria should be considered when judging feasibility: the required cost and expected value.

Types Of Feasibility Study

A feasibility analysis evaluates the project's potential for success; therefore, perceived objectivity is an essential factor in the credibility of the study for potential investors and lending institutions. There are five types of feasibility study—separate areas that a feasibility study examines, described below.

1. Technical Feasibility

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware, software, and other technical requirements of the proposed system. As an exaggerated example, an organization wouldn't want to try to put Star Trek's transporters in their building—currently, this project is not technically feasible.

2. Economic Feasibility

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility—helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

3. Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws. Let's say an organization wants to construct a new office building in a specific location. A feasibility study might reveal the organization's ideal location isn't zoned for that type of business. That organization has just saved considerable time and effort by learning that their project was not feasible right from the beginning.

3-LITERATURE SURVEY

3.1 Communication security in Internet of Thing: Preventive measure and avoid DDoS attack over IoT network:

https://dl.acm.org/doi/abs/10.5555/2872550.2872552

The idea of Internet of Things (IoT) is implanting networked heterogeneous detectors into our daily life. It opens extra channels for information submission and remote control to our physical world. A significant feature of an IoT network is that it collects data from network edges. Moreover, human involvement for network and devices maintenance is greatly reduced, which suggests an IoT network need to be highly self-managed and self-secured. For the reason that the use of IoT is growing in many important fields, the security issues of IoT need to be properly addressed. Among all, Distributed Denial of Service (DDoS) is one of the most notorious attacking behaviors over network which interrupt and block genuine user requests by flooding the host server with huge number of requests using a group of zombie computers via geographically distributed internet connections. DDoS disrupts service by creating network congestion and disabling normal functions of network components, which is even more disruptive for IoT. In this paper, a lightweight defensive algorithm for DDoS attack over IoT network environment is proposed and tested against several scenarios to dissect the interactive communication among different types of network nodes.

A look in the mirror: Attacks on package managers

https://www2.cs.arizona.edu/~jhh/papers/ccs08.pdf

Package managers are a privileged, centralized mechanism for software update and are essential to the security of modern computers. This work studies the security of ten popular package managers. These package managers use different mechanisms to provide security including signatures embedded in the package, signatures on metadata detached from the packages, or a signature on the root metadata (a file that contains the secure hashes of the package metadata). The security models used by these package managers are compared and contrasted. The threat model used to evaluate security in this paper is an attacker that controls a mirror (a copy of the main repository's contents for a distribution). We demonstrate that it is trivial for an attacker to control an official mirror for a popular distribution. An attacker can compromise a client who either installs software created by the attacker or installs an outdated version of a package with a vulnerability the attacker knows how to exploit. Furthermore, every package manager studied can be compromised by an attacker who controls a mirror without compromising a private key. In fact, 5 of the 10 package managers studied have security flaws that allow an attacker to compromise every client that requests a package from the mirror. We estimate that an attacker with a mirror that costs $50 per week could compromise between 150 and 1500 clients per week depending on the package manager. An existing package manager is modified to add a layered approach to security where multiple signatures are used. The updated package manager is evaluated in practical use. By using a layered approach to security, the package manager provides a high degree of usability and is not vulnerable to the attacks on existing package managers. The overhead of additional security mechanisms is 2-5% in practice and so should not be a deterrent. The purpose of this work is to not only point out security issues and provide solutions but also to raise an alarm to the imminent threat of attacks on package managers. Package managers are a weak point in the security of modern computers. Given the simplicity of compromising systems through package managers, developers and distributions must act quickly and intelligently to avert disaster.

Internet of Things: Features, challenges, and vulnerabilities

https://elvedit.com/journals/IJACSIT/wp-content/uploads/2015/02/internet-of-things.pdf

The terminology Internet of Things (IoT) refers to a future where every day physical objects are connected by the Internet in one form or the other, but outside the traditional desktop realm. The successful emergence of the IoT vision, however, will require computing to extend past traditional scenarios involving portables and smart-phones to the connection of everyday physical objects and the integration of intelligence with the environment. Subsequently, this will lead to the development of new computing features and challenges. The main purpose of this paper, therefore, is to investigate the features, challenges, and weaknesses that will come about, as the IoT becomes reality with the connection of more and more physical objects. Specifically, the study seeks to assess emergent challenges due to denial of service attacks, eavesdropping, node capture in the IoT infrastructure, and physical security of the sensors. We conducted a literature review about IoT, their features, challenges, and vulnerabilities. The methodology paradigm used was qualitative in nature with an exploratory research design, while data was collected using the desk research method. We found that, in the distributed form of architecture in IoT, attackers could hijack unsecured network devices converting them into bots to attack third parties. Moreover, attackers could target communication channels and extract data from the information flow. Finally, the perceptual layer in distributed IoT architecture is also found to be vulnerable to node capture attacks, including physical capture, brute force attack, DDoS attacks, and node privacy leaks.

Managing IoT devices using blockchain platform

https://ieeexplore.ieee.org/document/7890132

Since the start of Bitcoin in 2008[1], blockchain technology emerged as the next revolutionary technology. Though blockchain started off as a core technology of Bitcoin, its use cases are expanding to many other areas including finances, Internet of Things (IoT), security and such[2]. Currently, many private and public sectors are diving into the technology[3]. Aside from that, as software and hardware improve, we would see the beginning of IoT. And those IoT devices need to communicate and synchronize with each other. But in situations where more than thousands or tens of thousands of IoT devices connected, we expect that using current model of server-client may have some limitations and issues while in synchronization. So, we propose using blockchain to build IoT system. Using blockchain, we can control and configure IoT devices. We manage keys using RSA public key cryptosystems where public keys are stored in Ethereum and private keys are saved on individual devices. Specifically, we choose Ethereum as our blockchain platform because using its smart contract, we can write our own Turing-complete code to run on top of Ethereum. Thus, we can easily manage configuration of IoT devices and build key management system. Even though we can simply use account as a key management system, which most of blockchain platform supports, we decide to use Ethereum because we can manage the system in a more fine-grained way. For the proof of a concept, we use a few IoT devices instead of a full system of IoT system, which consists of thousands of IoT devices. But in our later study, we would like to build a fully scaled IoT system using blockchain.

Blockchain as a service for IoT

https://ieeexplore.ieee.org/document/7917130

A blockchain is a distributed and decentralized ledger that contains connected blocks of transactions. Unlike other ledger approaches, blockchain guarantees tamper proof storage of approved transactions. Due to its distributed and decentralized

organization, blockchain is beeing used within IoT e.g. to manage device configuration, store sensor data and enable micro-payments. This paper presents the idea of using blockchain as a service for IoT and evaluates the performance of a cloud and edge hosted blockchain implementation.

4.SYSTEM ANALYSIS

EXISTING SYSTEM

The project recognizes that traditional methods for updating IoT devices have several vulnerabilities and shortcomings. These vulnerabilities include the risk of malicious software updates, weak payment systems, potential interruptions in the update process, breaches of confidentiality during data transmission, the possibility of receiving invalid software updates, and the threat of rollback attacks. These vulnerabilities can significantly impact the security and reliability of IoT systems.

DISADVANTAGES OF EXISTING SYSTEM:

1. There have also been software repository state attacks against package managers that provide software libraries and components. Therefore, ensuring the confidentiality, integrity, and availability of the software update process is critical.
2. Furthermore, existing research assumes the manufacturer is honest by default and thus does not provide a guarantee of a valid software delivery transaction. In other words, the payment can occur without software delivery or vice-versa.
3. Last but not least, in compliance-driven industries such as health care, government, energy, and automotive, having proof of software update and installation for auditors is critical to the manufacturer keeping its business license by maintaining compliance with regulations

SYSTEM DESIGN

SYSTEM ARCHITECTURE:

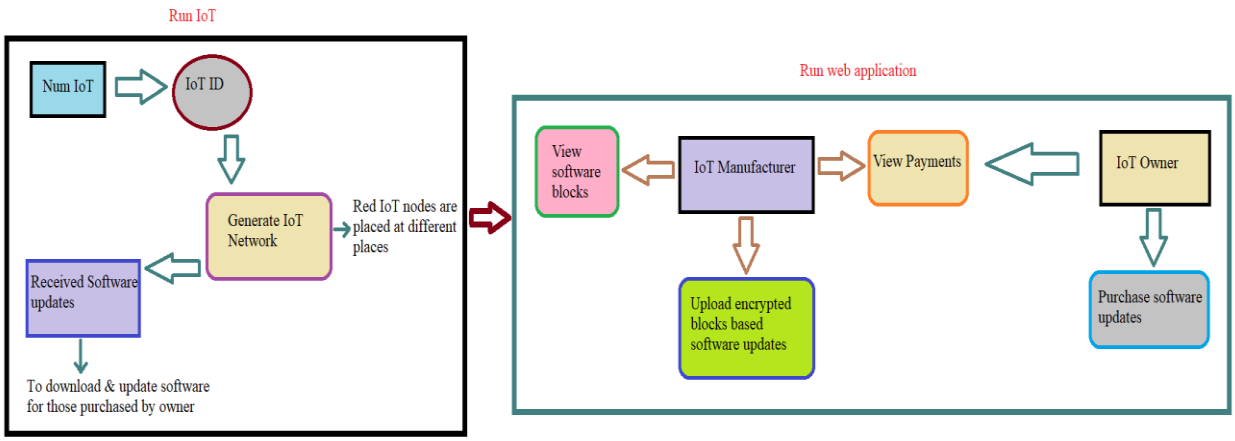


Fig.5.1.1 System architecture

DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that

depicts information flow and the transformations that are applied as data moves from input to output.

4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

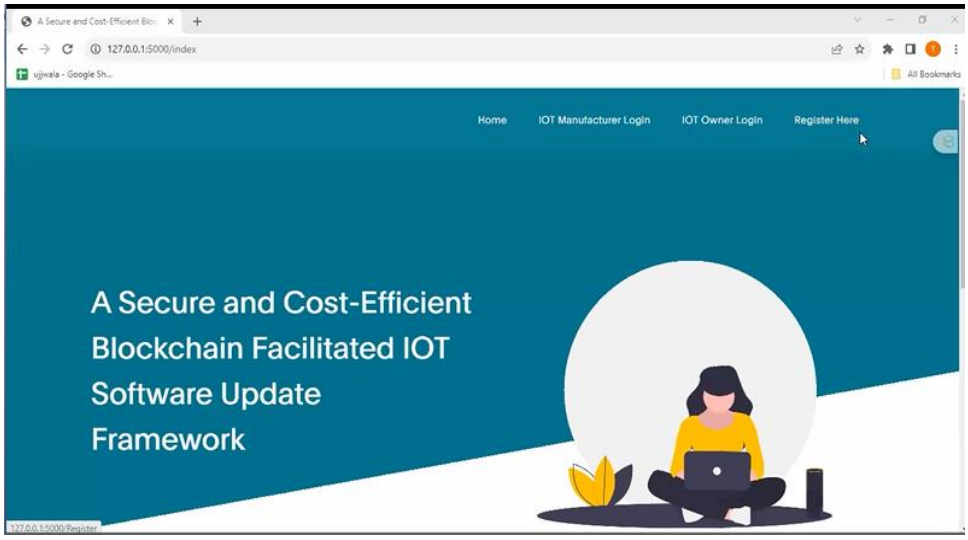
5. IMPLEMENTATION

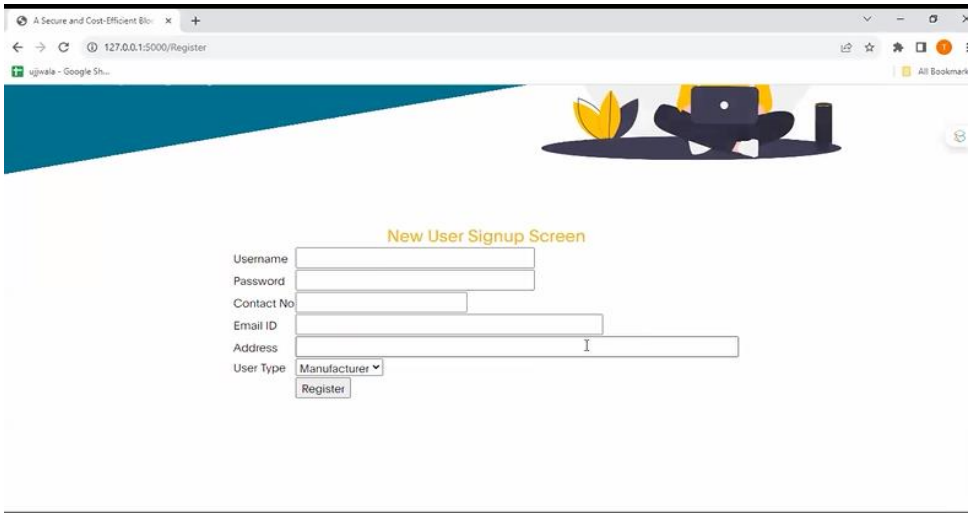
MODULES:

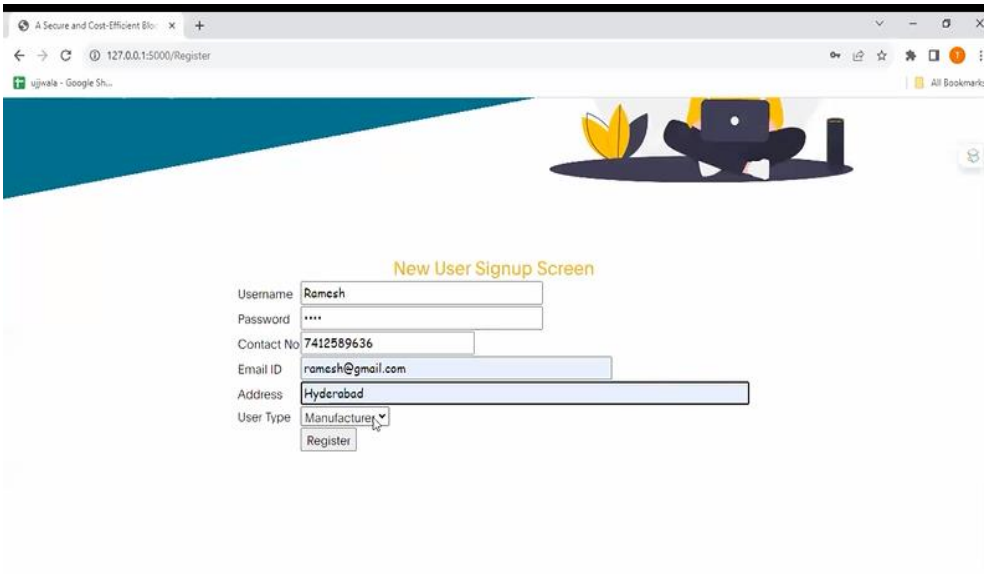
- **Register:** In this module, users can create an account. They need to provide basic information like a username and password. They can choose to register as either IoT device manufacturers or IoT device owners.
- **IOT Manufacturer Login:** Manufacturers can log in using their registered credentials. Once logged in, they can access features specific to manufacturers, such as uploading software updates.
- **Upload Encrypted Blocks Based Software Updates:** Within this module, manufacturers can upload software updates. These updates are divided into blocks, encrypted for security, and then stored. The location of the stored updates is recorded in the blockchain.
- **View Software Blocks**: Here, manufacturers can view the details of software updates, including how they are divided into blocks and encrypted. This provides transparency and control over the software update process.
- **View Payments:** This module allows manufacturers to review payment records related to software updates. It helps in tracking financial transactions and ensuring that payments are accurate.
- **IOT Owner Login:** IoT device owners can log in using their credentials. Once logged in, they can access features designed for owners, such as purchasing software updates.
- **Purchase Software Updates:** In this module, IoT device owners can browse and select software updates they want to install on their devices. They can make payments for these updates to ensure their devices have the latest software.
- **IOT Simulation:** As actual IoT devices are not available, the project employs a simulation-based IoT application. The simulation involves IoT devices connecting to the blockchain to download software updates.
- **Generate IOT Network:** Users can create simulated IoT networks, specifying the number of IoT devices and their configuration. This feature aids in testing the system's functionality.
- **Received Software Updates:** In this module, IoT devices receive and install the purchased software updates. This process ensures that devices are up-to-date with the latest software, enhancing their functionality and security.

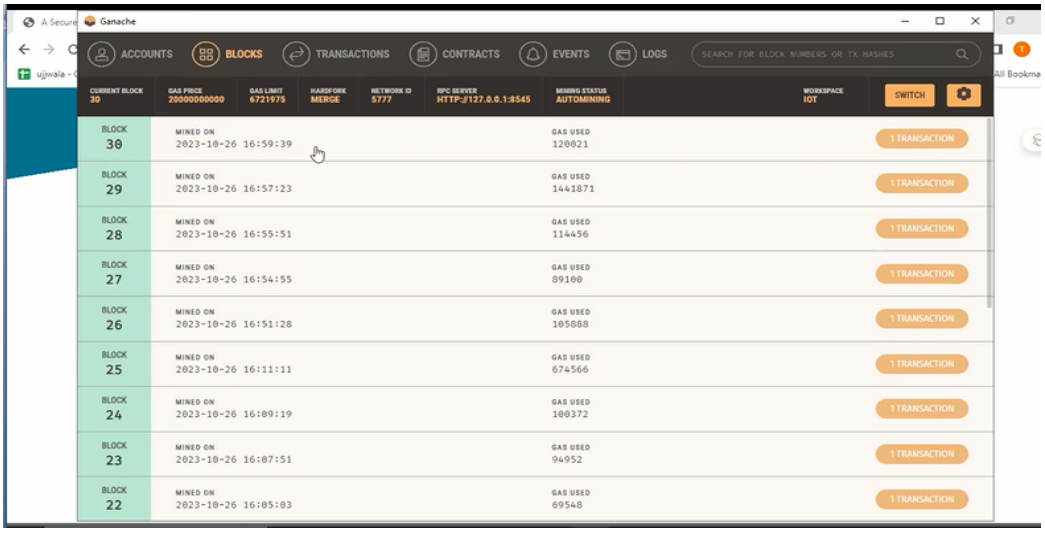
6-SCREENSHOTS

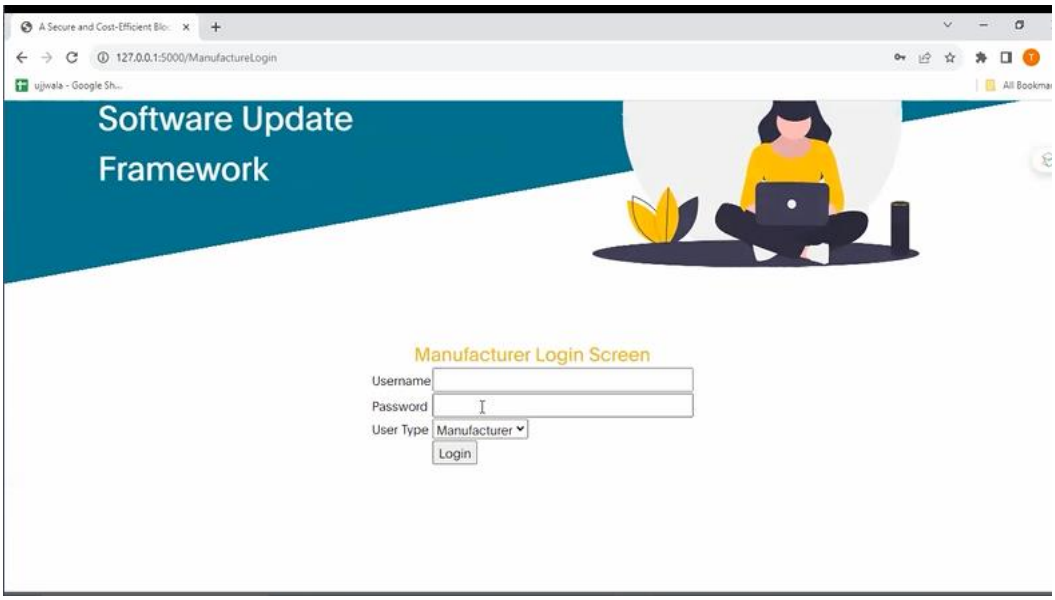
Output screens

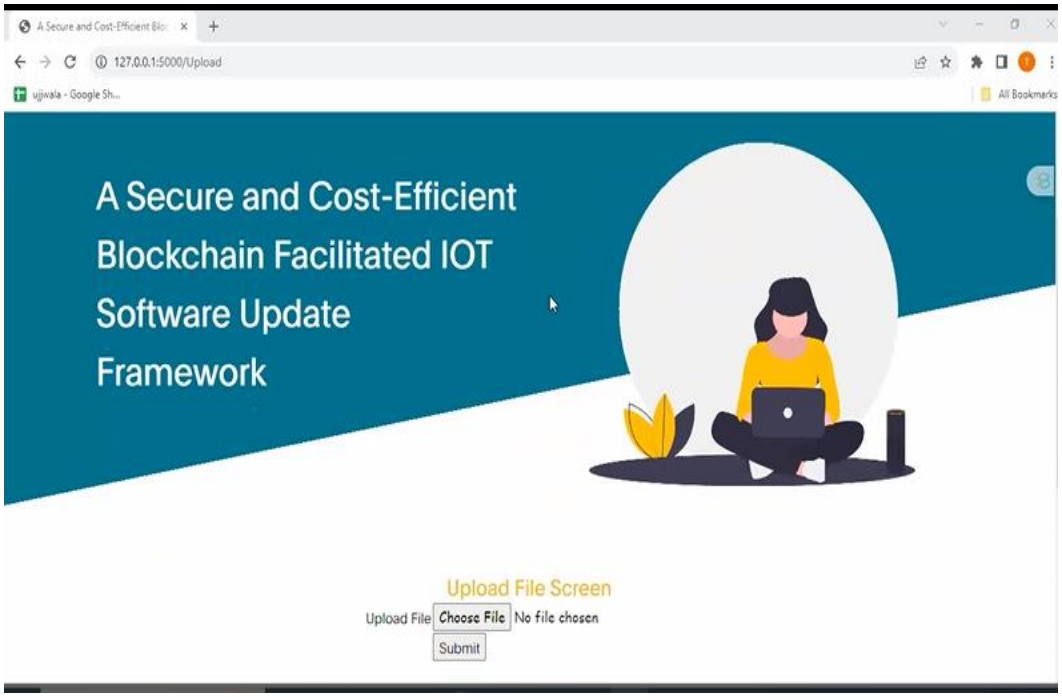

Home
IOT Manufacturer Login
IOT Owner Login
Register Here
A Secure and Cost-Efficient Blockchain Facilitated IOT Software Update Framework

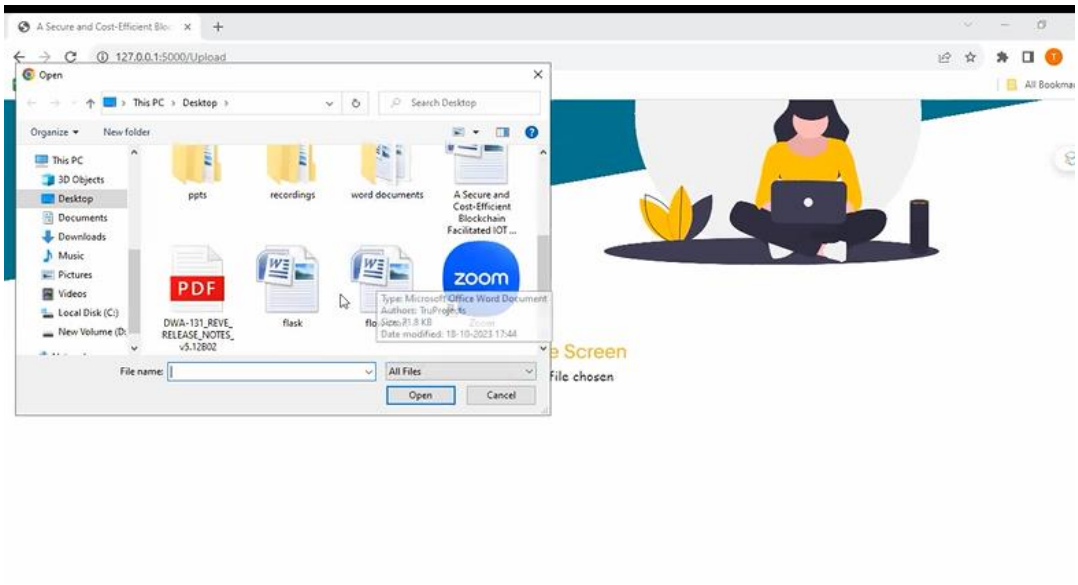

New User Signup Screen
Username
Password
Contact No
Email ID
Address
User Type
Manufacturer
Register

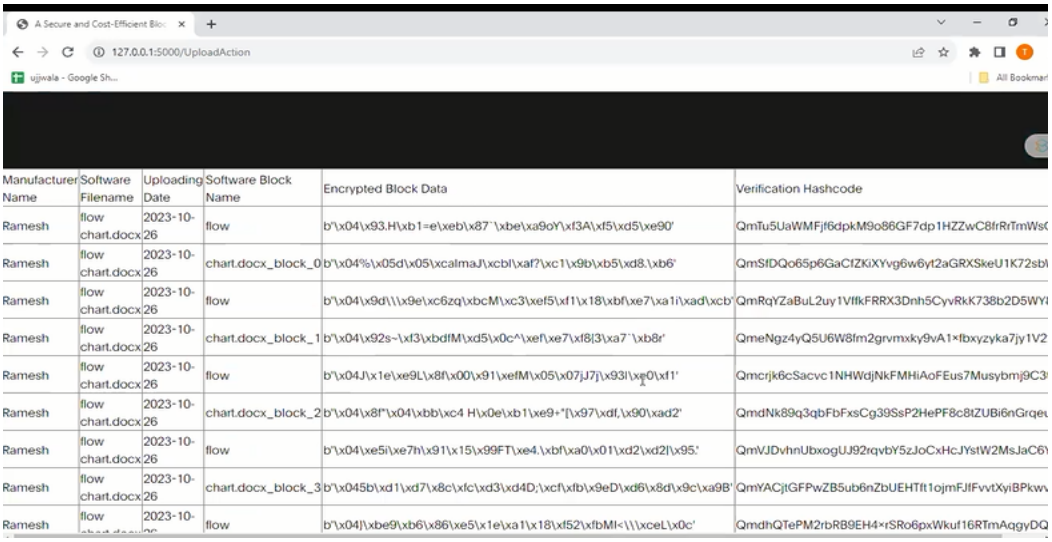

New User Signup Screen
Username
Ramesh
Password
Contact No
7412589636
Email ID
ramesh@gmail.com
Address
Hyderabad
User Type
Manufacturer
Register


Ganache
ACCOUNTS
BLOCKS
TRANSACTIONS
CONTRACTS
EVENTS
LOGS
SEARCH FOR BLOCK NUMBERS OR TX HASHES
SWITCH
BLOCK 30 MINED ON 2023-10-26 16:59:39 GAS USED 120021 1 TRANSACTION
BLOCK 29 MINED ON 2023-10-26 16:57:23 GAS USED 1441871 1 TRANSACTION
BLOCK 28 MINED ON 2023-10-26 16:55:51 GAS USED 114456 1 TRANSACTION
BLOCK 27 MINED ON 2023-10-26 16:54:55 GAS USED 89100 1 TRANSACTION
BLOCK 26 MINED ON 2023-10-26 16:51:28 GAS USED 105888 1 TRANSACTION
BLOCK 25 MINED ON 2023-10-26 16:11:11 GAS USED 674566 1 TRANSACTION
BLOCK 24 MINED ON 2023-10-26 16:09:19 GAS USED 100372 1 TRANSACTION
BLOCK 23 MINED ON 2023-10-26 16:07:51 GAS USED 94952 1 TRANSACTION
BLOCK 22 MINED ON 2023-10-26 16:05:03 GAS USED 69548 1 TRANSACTION

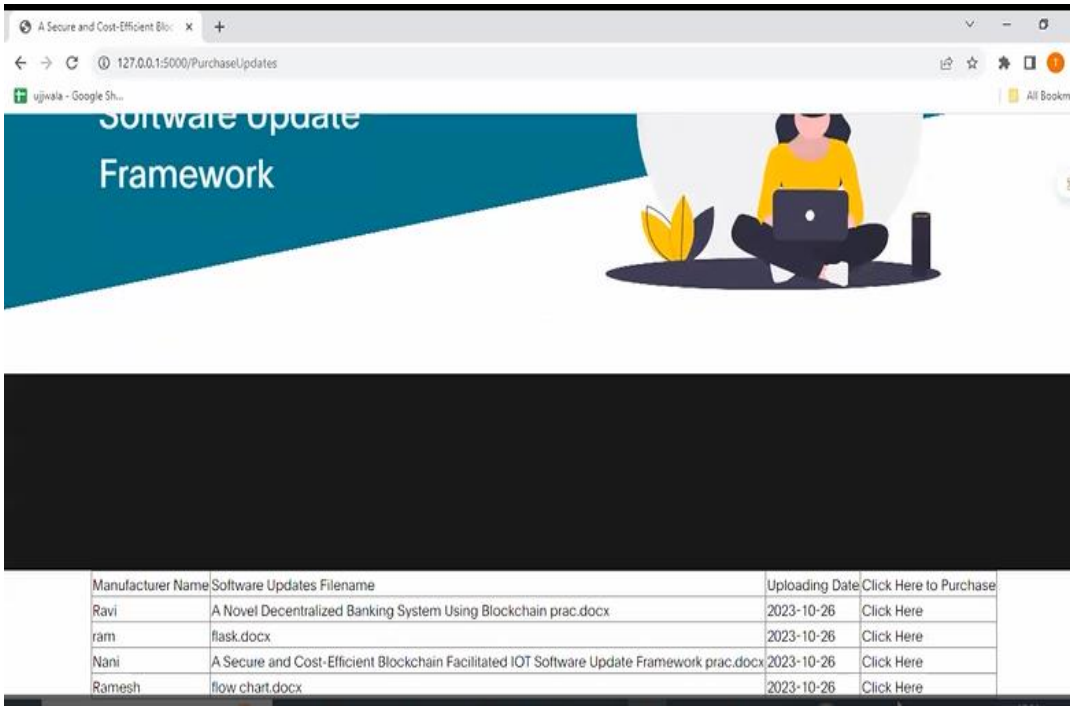

127.0.0.1:5000/ManufactureLogin
Software Update
Framework
Manufacturer Login Screen
Username
Password
User Type
Manufacturer
Login


A Secure and Cost-Efficient Blockchain Facilitated IOT Software Update Framework
Upload File Screen
Upload File
Choose File
No file chosen
Submit


Open
This PC
Desktop
Search Desktop
Organize
New folder
ppts
recordings
word documents
zoom
PDF
flask
File name:
All Files
Open
Cancel



Manufacturer Name	Software Filename	Uploading Date	Software Block Name	Encrypted Block Data	Verification Hashcode
Ramesh	flow chart.docx	2023-10-26	flow	b'\x04\x93.H\xb1=e\xeb\x87`\xbe\xa9oY\xf3A\xf5\xd5\xe90'	QmTu5UaWMFjf6dpkM9o86GF7dp1HZZwC8frRrTmWs
Ramesh	flow chart.docx	2023-10-26	chart.docx_block_0	b'\x04%\x05d\x05\xcalmaJ\xcbl\xaf?\xc1\x9b\xb5\xd8.\xb6'	QmSfDQo65p6GaCfZKiXYvg6w6yt2aGRXSkeU1K72sb
Ramesh	flow chart.docx	2023-10-26	flow	b'\x04\x9d\\\x9e\xc6zq\xbcM\xc3\xef5\xf1\x18\xbf\xe7\xa1i\xad\xcb'	QmRqYZaBuL2uy1VffkFRRX3Dnh5CyvRkK738b2D5WY
Ramesh	flow chart.docx	2023-10-26	chart.docx_block_1	b'\x04\x92s~\xf3\xbdfM\xd5\x0c^\xef\xe7\xf8|3\xa7`\xb8r'	QmeNgz4yQ5U6W8fm2grvmxky9vA1×fbxyzyka7jy1V2
Ramesh	flow chart.docx	2023-10-26	flow	b'\x04J\x1e\xe9L\x8f\x00\x91\xefM\x05\x07jJ7j\x93l\xe0\xf1'	Qmcrjk6cSacvc1NHWdjNkFMHiAoFEus7Musybmj9C3
Ramesh	flow chart.docx	2023-10-26	chart.docx_block_2	b'\x04\x8f*\x04\xbb\xc4 H\x0e\xb1\xe9+*[\x97\xdf,\x90\xad2'	QmdNk89q3qbFbFxsCg39SsP2HePF8c8tZUBi6nGrqe
Ramesh	flow chart.docx	2023-10-26	flow	b'\x04\xe5i\xe7h\x91\x15\x99FT\xe4.\xbf\xa0\x01\xd2\xd2l\x95.'	QmVJDvhnUbxogUJ92rqvbY5zJoCxHcJYstW2MsJaC6
Ramesh	flow chart.docx	2023-10-26	chart.docx_block_3	b'\x045b\xd1\xd7\x8c\xfc\xd3\xd4D;\xcf\xfb\x9eD\xd6\x8d\x9c\xa9B'	QmYACjtGFPwZB5ub6nZbUEHTft1ojmFJfFvvtXyiBPkwv
Ramesh	flow	2023-10-26	flow	b'\x04J\xbe9\xb6\x86\xe5\x1e\xa1\x18\xf52\xfbMI<\\\xceL\x0c'	QmdhQTePM2rbRB9EH4×rSRo6pxWkuf16RTmAqgyDQ



Manufacturer Name	Software Updates Filename	Uploading Date	Click Here to Purchase
Ravi	A Novel Decentralized Banking System Using Blockchain prac.docx	2023-10-26	Click Here
ram	flask.docx	2023-10-26	Click Here
Nani	A Secure and Cost-Efficient Blockchain Facilitated IOT Software Update Framework prac.docx	2023-10-26	Click Here
Ramesh	flow chart.docx	2023-10-26	Click Here

7-CONCLUSION

- The project effectively mitigated vulnerabilities in traditional IoT software updates, ensuring improved security and reliability. This was achieved through the use of blockchain, CPABE, and ECDSA technologies.
- Blockchain technology played a pivotal role by providing tamper-proof data security, guaranteeing software update integrity, and facilitating transparent payment handling. This ensured trust and reliability in transactions.
- The incorporation of CPABE reduced computational overhead in key generation, simplifying access control. This efficiency improved overall system performance.
- Utilizing IPFS storage not only proved cost-effective but also enhanced security by distributing software updates into blocks at different IPFS locations. This innovative approach mitigated vulnerabilities associated with centralized storage.
- The project effectively integrated Ganache, enhancing the development and testing of IoT

software updates. It provided a user-friendly interface for monitoring blockchain activities, contributing to project security and reliability.

- The project's scalable design, validated through simulation, allows it to adapt to larger IoT networks and diverse applications. It demonstrated successful software update reception and application, affirming its functionality and effectiveness.

REFERENCES

[1] S. Poslad, ‘‘Ubiquitous computing: Basics and vision,’’ in Ubiquitous Computing: Smart Devices, Environments and Interactions. Hoboken, NJ, USA: Wiley, 2011, pp. 1–40.

[2] F. Wortmann and K. Fluchter, ‘‘Internet of Things,’’ Bus. Inf. Syst. Eng., vol. 57, no. 3, pp. 221–224, 2015.

[3] (Jun. 2019). Global Internet of Things (IoT) Market Size and Forecast To 2026. [Online]. Available: https://www.verifiedmarketresearch. com/product/global-internet-of-things-iot-market-size-and-forecast-to- 2026/

[4] (Dec. 2020). Global Internet of Things (IoT) Market By Software Solution, Report ID 6403. [Online]. Available: https://www.verifiedmarketresearch. com/product/global-internet-of-things-iot-market-size-and-forecast-to- 2026/

[5] (Jan. 2020). Internet of Things (IoT) in the US. [Online]. Available: https://www-statista-com.libproxy.scu.edu/study/61733/internet-ofthings-iot-in-the-us/

[6] (Jan. 2020). Size of the Internet of Things (IoT) in Retail Market in the United States From 2014 to 2025. [Online]. Available: https://wwwstatista-com.libproxy.scu.edu/statistics/688756/iot-in-retail-market-inthe-us/

[7] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, ‘‘DDoS in the IoT: Mirai and other Botnets,’’ Computer, vol. 50, no. 7, pp. 80–84, 2017.

[8] L. H. Newman. (2016). The Botnet That Broke the Internet Isn’t Going Away. [Online]. Available: https://www.wired.com/2016/12/botnet-brokeinternet-isnt-going-away/

[9] C. Zhang and R. Green, ‘‘Communication security in Internet of Thing: Preventive measure and avoid DDoS attack over IoT network,’’ in Proc. 18th Symp. Commun. Netw., 2015, pp. 8–15.

[10] Newman. (Oct. 2016). What We Know About Friday’s Massive East Coast Internet Outage. [Online]. Available: https://www.wired.com/ 2016/10/internet-outage-ddos-dns-dyn/

[11] (Mar. 2016). Inside the Cunning, Unprecedented Hack of Ukraine’s Power Grid. [Online]. Available: https://www.wired.com/2016/03/insidecunning-unprecedented-hack-ukraines-power-grid/

[12] (Mar. 2016). Analysis of the Cyber Attack on the Ukrainian Power Grid: Defense Use Case. [Online]. Available: https://media. kasperskycontenthub.com/wp-content/uploads/sites/43/2016/05/2008 1514/E-ISAC_SANS_Ukraine_DUC_5.pdf

[13] J. Cappos, J. Samuel, S. Baker, and J. H. Hartman, ‘‘A look in the mirror: Attacks on package managers,’’ in Proc. 15th ACM Conf. Comput. Commun. Secur., New York, NY, USA, Oct. 2008, pp. 565–574.

[14] E. Alsaadi and A. Tubaishat, ‘‘Internet of Things: Features, challenges, and vulnerabilities,’’ Int. J. Adv. Comput. Sci. Inf. Technol., vol. 4, no. 1, pp. 1–13, 2015.

[15] S. Huh, S. Cho, and S. Kim, ‘‘Managing IoT devices using blockchain platform,’’ in Proc. 19th

Int. Conf. Adv. Commun. Technol. (ICACT), 2017, pp. 464–467.

[16] M. Samaniego and R. Deters, ‘‘Blockchain as a service for IoT,’’ in Proc. IEEE Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData), 2016, pp. 433–436.

[17] M. Samaniego and R. Deters, ‘‘Using blockchain to push software-defined IoT components onto edge hosts,’’ in Proc. Int. Conf. Big Data Adv. Wireless Technol., Nov. 2016, pp. 110–119.

[18] D. Li, R. Du, Y. Fu, and M. H. Au, ‘‘Meta-key: A secure data-sharing protocol under blockchain-based decentralized storage architecture,’’ IEEE Netw. Lett., vol. 1, no. 1, pp. 30–33, Mar. 2019.

[19] A. Dorri, S. S. Kanhere, and R. Jurdak, ‘‘Towards an optimized BlockChain for IoT,’’ in Proc. 2nd Int. Conf. Internet-Things Design Implement., Apr. 2017, pp. 173–178.

[20] T. Placho, C. Schmittner, A. Bonitz, and O. Wana, ‘‘Management of automotive software updates,’’ Microprocessors Microsyst., vol. 78, Oct. 2020, Art. no. 103257.