SELF-SECURE FIRMWARE MODEL FOR BLOCKCHAIN-ENABLED IOT ENVIRONMENT TO EMBEDDED SYSTEM

Narender Chinthamu[1], M. Prasad[2], Amit Jaykumar Chinchawade[3], Kazi Kutubuddin Sayyad Liyakat[4], Kakarla Deepti[5], Manideep Karukuri[6], Ch Manohar Kumar[7]

Article History: **Received:** 12.12.2022 **Revised:** 29.01.2023 **Accepted:** 15.03.2023

Abstract

Integrated hardware will be heavily utilized in Internet of Things (IoT) settings. Without the input of consumers, the smallest IoT systems will function and interact with one another, but their functions have to be accurate and secure from multiple threats. In this research, we concentrate on a key security concern for embedded devices in an IoT ecosystem and safe software updates. It is proposed to use blockchain technology in the latest firmware update method to safely verify a firmware version, verify the accuracy of firmware, and install the most recent firmware for embedded devices. The proposed system involves an embedded system asking nodes in a blockchain network for a software update and then receiving an answer to see if the firmware is existing or not. The embedded device receives the most recent firmware from a peer-to-peer node network if it is not already installed. The firmware's authenticity, or accuracy of firmware, is verified even if the version is the most recent. The proposed method ensures that the firmware of the embedded system is existing and unaltered. Thus, attacks that target well-known flaws in the software of embedded devices are neutralized.

Keywords: Embedded devices; IoT; Blockchain technology; firmware

[1]MIT (Massachusetts Institute of Technology) CTO Candidate, Enterprise Architect
[2]Assistant Professor Senior, School of Computer Science and Engineering, Vellore Institute of Technology (Chennai Campus), Chennai, TamilNadu - 600127.
[3]Assistant Professor, Department of Electronics and Computer Engineering, Sharad Institute of Technology College of Engineering, Yadrav (Ichalkaranji). Maharashtra,India,
[4]Professor, Department of Computer Science and Engineering Brahmdevdada Mane Institute of Technology, Solapur, Maharashtra, India ,
[5]Assistant Professor,Department of Electronics & Communication Engineering, Vasavi college of Engineering, Ibrahimbagh,Hyderabad - 500 031,Telangana, India
[6]University of Texas at Arlington, MSBA Graduate, DALLAS, TEXAS United States,
[7]Assistant Professor, Gayatri Vidya Parishad college for Degree and PG Courses (A), Rushikonda, Visakhapatnam-530045, Andhrapradesh ,India,

Email: [1]narender.chinthamu@gmail.com, [2]prasad.psdm@gmail.com, [3]amitchinchawade@sitcoe.org.in, [4]drkkazi@gmail.com, [5]deepti@staff.vce.ac.in, [6]manideepkarukoori@gmail.com, [7]manohar.chebrolu@gmail.com

DOI: 10.31838/ecb/2023.12.s3.075

1. Introduction

Blockchain and the IoT would both be building blocks of the vision of a linked and more robotic society. Interesting application scenarios and related issues are produced when these two fields converge [1-2]. The fact that blockchain offers a workable answer to the problem of a shortage of reliable mechanisms to regulate trustworthiness and the sharing of actual data among IoT devices is a persuasive factor driving the synergies between such two fields [3]. The fact that blockchain technology is still recent and has not been widely adopted throughout various businesses is a significant hurdle in this field.

Blockchain enables verified untraceable payments by numerous marketing, append-only accessible ledgers. In which buyers and sellers are linked throughout a reliable third party, generally a bank or broker in financial operations is a significant issue that blockchain overcomes [4]. The majority of operations right now use a centralized paradigm. With the use of blockchains, we can eliminate this middleman and go from a centralized to a decentralized system. Through the blockchain, each participant can confirm the transactions of all other participants [5]. The IoT is a huge network that includes trillions of devices, some of which may seek to interact with one another. The synchronization of millions of devices presents a significant problem in this situation [6]. Blockchain provides a decentralized approach because a centralized paradigm is not appropriate in this situation.

The accessibility and expansion of cloud applications, which can save and handle the huge amounts of information produced by IoT systems, is one of the factors driving the IoT market. This is so that IoT systems and the cloud may work together effectively [7]. The cloud has memory, computing, and connectivity that are many orders of magnitude more than those of IoT devices. For example, the cloud can store at least 2.5 Exabytes of data created every day, while an Amazon Blink Security video has very limited storage [8–10]. A modern computing discipline known as CloudIoT has emerged as a consequence of the interaction between the cloud and connected devices [11].

All information on each device makes it simpler to retrieve the information, which is especially helpful for aircraft applications. It is possible to send an escort to a single agent, pick it up, and bring it back to earth without keeping the remainder of the system in place [12]. Just after the operation has started, more representatives can be incorporated into the system, and if the robot's control method is similarly decentralized, the cluster will accept these and modify their behavior to take advantage of the new assets [13]. An option would be to use a "sniffer agent" solely to obtain an existing blockchain.

2. Related works

The blockchain-based data architecture makes it possible to identify errors and harmful activity because it serves as an unquestionable source of accuracy within a system. A node can fail as a result of a failure that results from malicious intent, malfunctioning software, possibly flawed information, or an internal node mistake. In any case, agreements can be used to localize errors at the system level [14–16]. Smart contracts serve as the link between such nodes and the blockchain network in permissionless blockchain networks like Fabric. A smart contract can validate network signals coming from a node and determine whether they are authentic. A smart contract is capable of recording the intended connection with the variables for verification purposes and not running a flawed program when it detects information that goes beyond a permitted set of variables [17]. Destiny and strong identification are two additional requirements for high availability, both of which are by necessity met by any permissionless blockchain system. According to predictability, any node that receives the identical input must produce the exact result. The requirement for individuals to be individually recognizable is a clear identity.

Usually, asymmetrical cryptographic techniques like ECC or RSA are employed to secure the distant firmware upgrade. To illustrate, the communication breakdown over the entire firmware record is generated with a hashing algorithm that can provide authenticity and authenticity of the firmware, and the decoder is then validated using a secret key [18]. The firmware package has the created digital certificate and the associated public key and private key. Before performing the firmware upgrade itself, a security screening procedure is carried out after acquiring the firmware file from the source, in which the digital signature is only checked using the public key that is connected [19–20]. The destination embedded device's real firmware upgrade is initiated after the security screening procedure has been finished. Due to the client-server architecture used in the present firmware distribution system, significant internet traffic may result from simultaneous requests for firmware upgrade packages from embedded systems.

3. Proposed work

The attribute-based encryption automatic update system permits an embedded device to ask blockchain networks for a default installation and then receive a reply from a node to learn whether or not the device's firmware is existing [21]. If the firmware is out-of-date, a meta-document including

a neighbor listing to get the most recent version is sent to the device so that it can retain the most recent version by obtaining it from a peer-to-peer firmware network made up of blockchain nodes. The peer-to-peer firmware sharing system can be developed using BitTorrent, it should be noted. If the firmware is existing, the blockchain servers are used to verify the firmware's authenticity. The proposed system's general architecture is depicted in Figure 1.

.

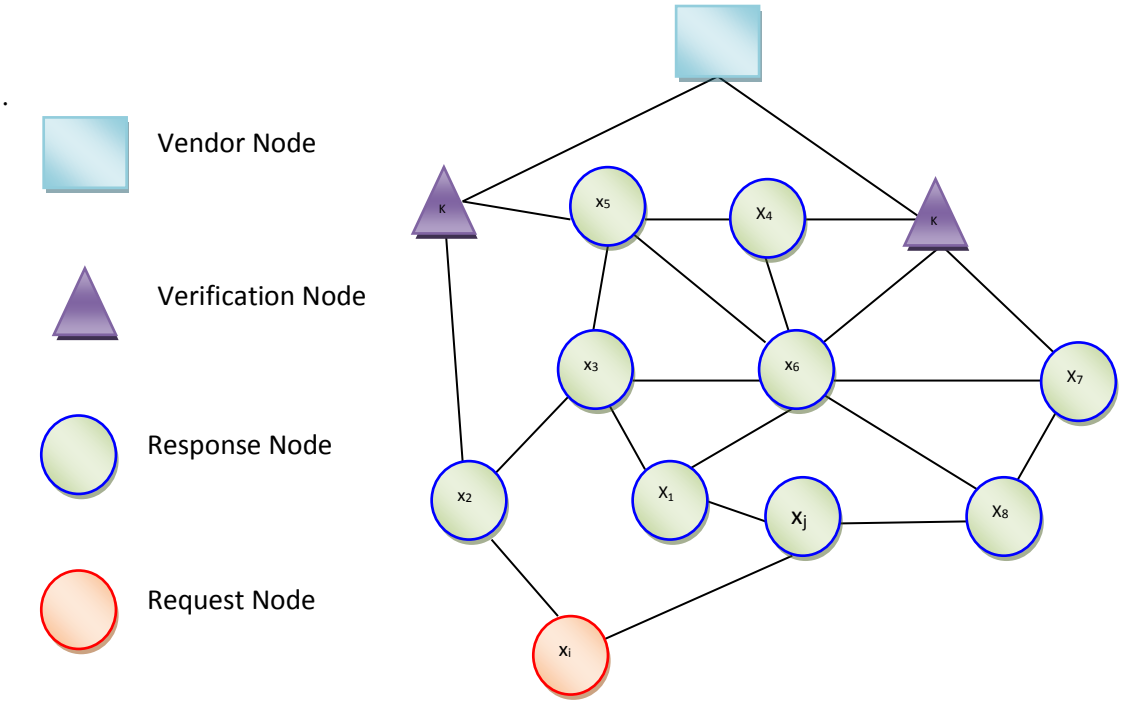


Figure 1 Overall architecture

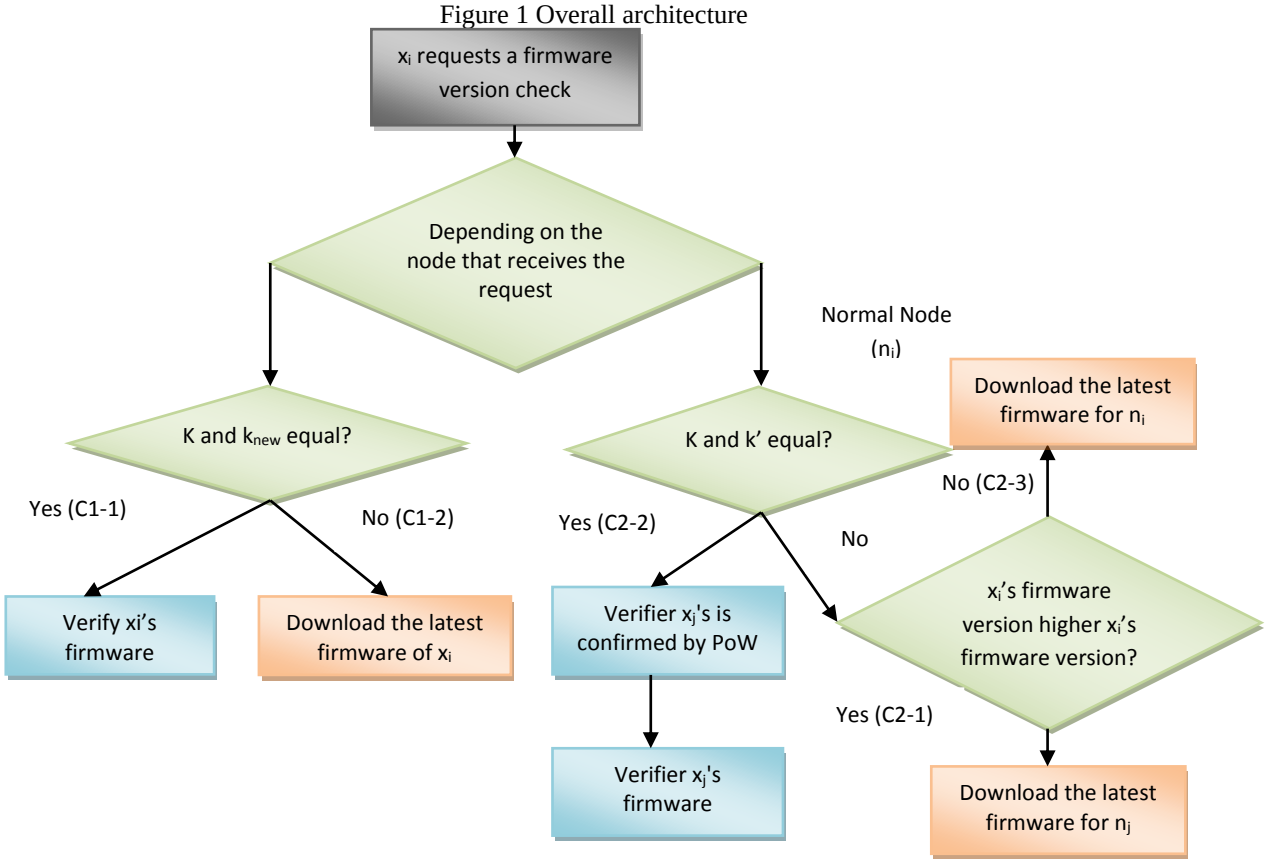
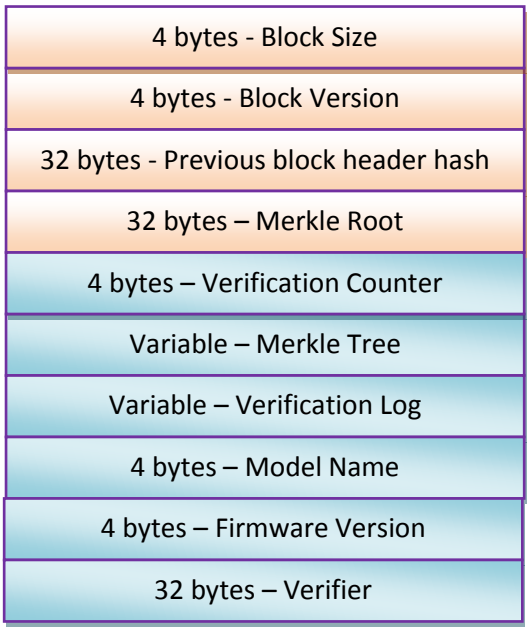


Figure 2: Flow graph of the proposed system

The proposed program's entire technique is depicted in Figure 2. In the proposed method, an embedded device, which is a typical node in a blockchain network, transforms into a requesting node when it broadcasts a version-check query message to initiate its firmware upgrade process. Additional regular nodes, such as reply networks and validation nodes, reply to the version-check post request after it has been disseminated throughout the blockchain network. The two scenarios are where the modifier differs based on the kind of node: Responses C1 and C2 come from the validation network vi and the reply nodes n j, respectively, to the demand node ni.

A validation node in C1 determines if a requesting node has the most recent firmware version or not. The verification node additionally verifies the validity of the requesting node's firmware if it has the most existing edition. If not, the firmware of the requesting node must be upgraded using the firmware to keep giving. A reply node in C2 matches its firmware version with the query node's firmware edition [22]. The response node requests other nodes in the blockchain network to validate the firmware file's hash code, which is termed a validator and will be detailed shortly if the firmware version of the replying node and the query node is identical. The response node considers that its firmware is right if the validator of the reply network is verified by other networks, which will be accomplished through six confirmations, or proof-of-work, of the blockchain. The reply node can then compare each other's validators to see if the request node's software has been changed. If the firmware versions of the demand and reply nodes disagree, the replying node additionally determines whether the demand node's firmware edition is more recent than its firmware version.

When contrasted to the Bitcoin blockchain, the proposed approach employs a unique structure. The block header and validation feature are what make the blocks in the proposed system, as seen in Figure 3. The frame, edition, original plaintext header hashing, and Merkle root make up the block header. The product identifier, firmware edition, certification validator, Merkle tree, confirmation log, and confirmation counter make up the certification area.



4. Results and discussions

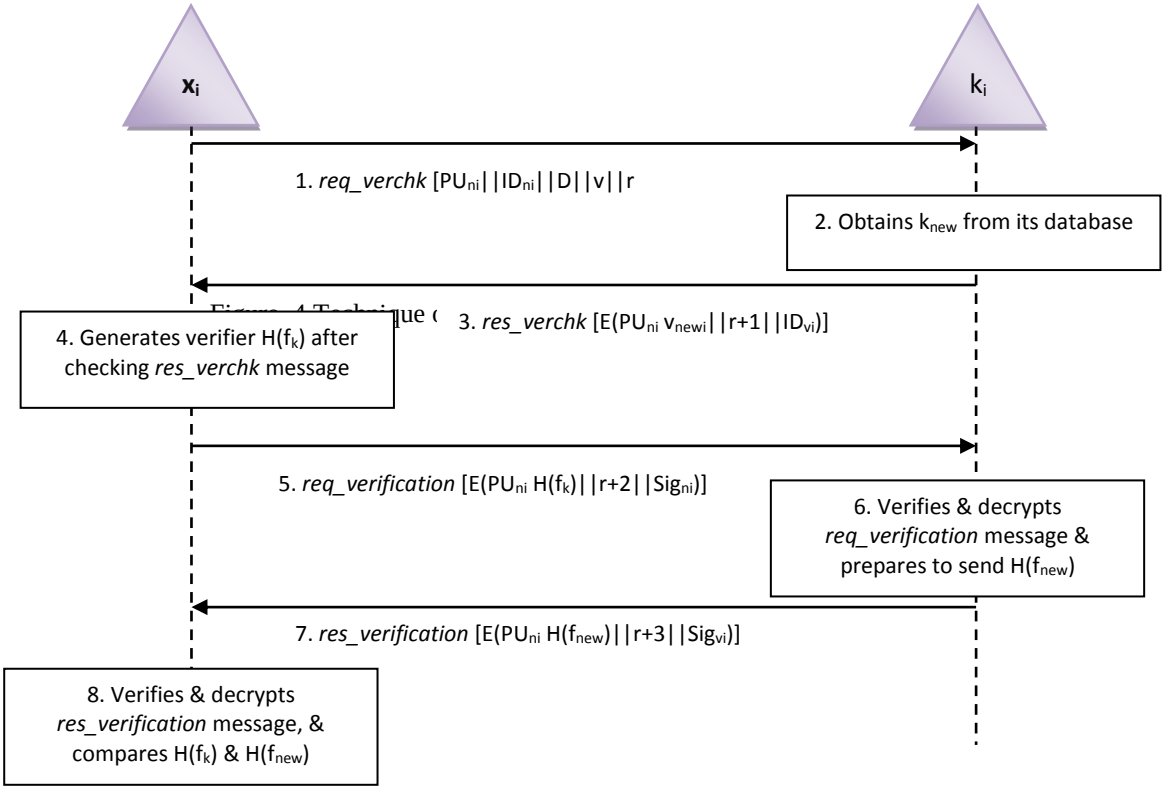
The two distinct operation situations in the proposed protocol are C1 and C2, as was already indicated. Then instance C1 is split into two particular things, and instance C2 is likewise split into instances C2-1, C2-2, and C2-3. The scenarios vary depending on the kind of reply node that gets the message requesting a version test that was sent by the requesting network.

The technique of C1-1 exposed in Figure. 4 is as follows.

1. To a public blockchain, a requesting node ni transmits a reqverchk message that includes PUni, i, Dni, D, v, and r.
2. Depending on i, Dni, D, and v, a validation network vi retrieves the most recent version number of ni from its storage when it receives the notification.

3. In response, vi transmits a message that contains the data vnew, r+1, i, Dvi and is encoded with 4. After decrypting the signal with PRni and performing r + 1 check, ni receives the information. When the newest firmware is present and v and vnew are equivalent, ni produces its validator H. (f v). If not, a fatal error is thrown and the procedure is ended.
5. After receiving the communication, ni checks its legitimacy and origination using the discussed PUni.

Sigvi and PUvi. The data is then decrypted using PRni and checksr +3. Ni also checks H(f v) and H(f vnew) to verify the accuracy of its firmware. If both validators agree, the process is deemed accomplished. Instead, ni must use a peer-to-peer firmware that provides a C1-2 to obtain the most recent software.



1. To a public blockchain, a requesting node ni transmits a reqverchk message includes PUni, i Dni, D, v, and r.
2. Depending on I Dni, D, and v, a validation node vi retrieves the most recent version number of ni out of its storage when it gets the message.
3. Vi answers by delivering a PUni-encrypted res verchk signal that contains vnew, Mvnew, r + 1, and i, Dvi.
4. After decrypting the communication with PRni and performing a r + 1 check, niobtains the information. Therefore, using Mvnew, ni gets H(f vnew) as shown in Figure 5.
5. After receiving the communication, ni checks its legitimacy and origination using the shared information Sigvi and PUvi. It then uses PRni to decipher the message and tests r + 3. The most recent firmware version can now be downloaded and installed by ni via a peer-to-peer firmware network system.

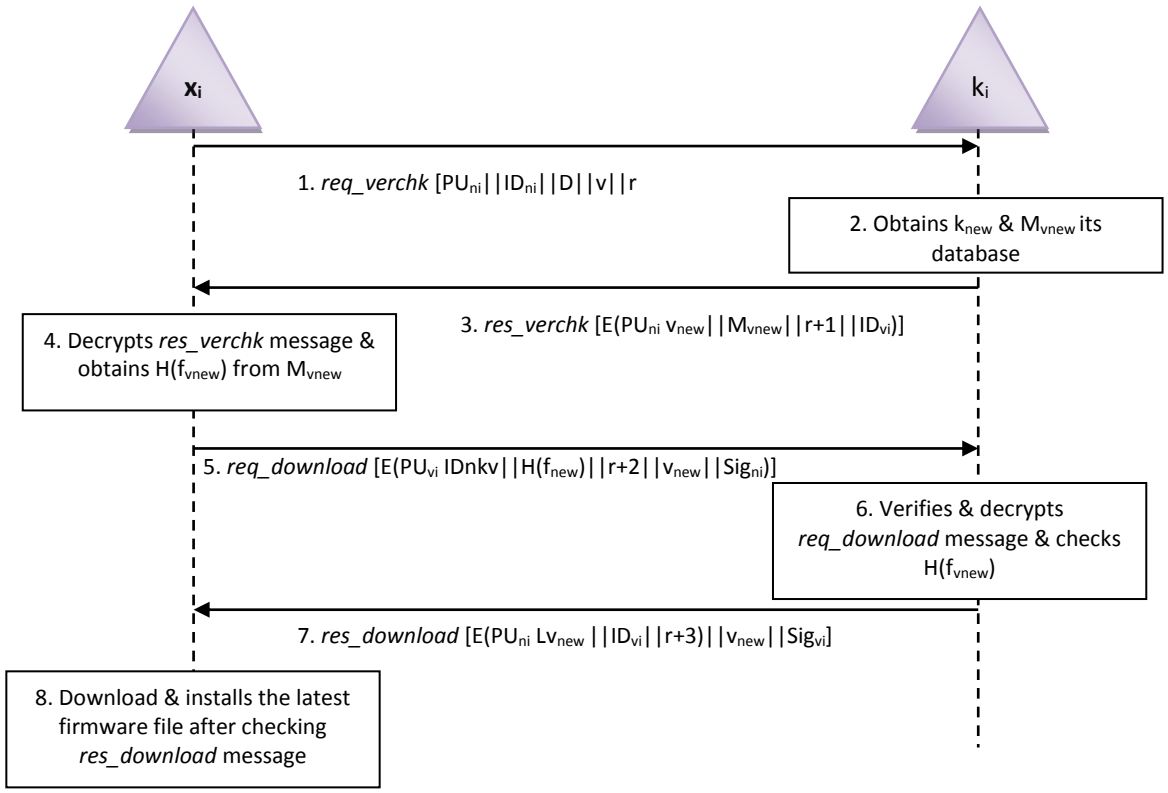


Figure 5: C1-2 technique: ni is not running the most recent software

By allowing an embedded system to immediately verify its firmware and acquire the most recent firmware if necessary, the proposed technique aims to reduce the attack window time. As a result, it aids in reducing the impact of attacks on recognized embedded electronic firmware flaws. All of the activities, including firmware verification and distribution to upgrade, are built on distributed modeling since this method was created for the IoT context. In many other words, BitTorrent has been employed as a peer-to-peer firmware network for obtaining the most recent software, while blockchain technology has indeed been employed for safe firmware version verification and authentication. Furthermore, in the peer-to-peer firmware network system, a validation node serves as a monitor. Thus, the validation node is viewed as a service that facilitates peer-to-peer connectivity for firmware installation. Using hashed tables, or putting in place a trackless Bit Torrent network, can enhance the proposed strategy.

The public key of a validation node is presumed to be pre-shared with all other nodes in the present scheme. The price of key distribution and management rises as the number of verification units does. However, in reality, there wouldn't be a huge number of validation nodes. The proposed technique allows an embedded system to maintain firmware consistency, or firmware accuracy while keeping it up to date. It does not imply, though, that the delivered firmware is not flawed or prone to problems like stack exchange. Additionally, the proposed approach does not ensure that the nodes concerned operate properly. We've assumed that every node functions by the planned design. In actuality, an accessibility and software alteration assault might affect certain systems.

The edition request message, which is a broadcasted communication, is sent by a query node to begin the firmware upgrade demand in the proposed schemes. The proposed technique thus has two distinct operating situations, such as C1 and C2, based on the node category that gets the wireless signal. However, the proposed methodology may result in extra network traffic and devices because of the structure of the radio transmitter. Blockchain technology and several fundamental aspects of asymmetric cryptography, such as transaction signing and encrypted, are used to secure the developed model.

5. Conclusion

In this work, we introduced a new firmware modernization technique that allows embedded devices operating in an IoT ecosystem to acquire the most recent firmware while safely verifying the firmware's accuracy and checking its edition. Although Bit Torrent can be utilized to construct a peer-to-peer firmware sharing network for firmware downloads, the proposed system depends on blockchain technology for firmware testing and verification. The proposed system's design and intricate operating processes have been described. Additionally, we have spoken about the advantages and disadvantages of the proposed work. As previously noted, the existing system has several restrictions and constraints, thus we will enhance the proposed system to address these issues and increase safety and adaptability.

6. References

Frikha, T., Chaabane, F., Aouinti, N., Cheikhrouhou, O., Ben Amor, N., &Kerrouche, A. (2021). Implementation of blockchain consensus algorithm on embedded architecture. Security and Communication Networks, 2021.

Qureshi, J. N., Farooq, M. S., Abid, A., Umer, T., Bashir, A. K., &Zikria, Y. B. (2022). Blockchain Applications for the Internet of Things: Systematic Review and Challenges. Microprocessors and Microsystems, 104632.

Pouraghily, A., & Wolf, T. (2019, February). A lightweight payment verification protocol for blockchain transactions on IoT devices.In 2019 International Conference on Computing, Networking, and Communications (ICNC) (pp. 617-623).IEEE.

Rao, A. R., & Dave, R. (2019, March). Developing hands-on laboratory exercises for teaching STEM students the internet-of-things, cloud computing and blockchain applications. In 2019 IEEE Integrated STEM Education Conference (ISEC) (pp. 191-198). IEEE.

T. P. Latchoumi, R. Swathi, P. Vidyasri and K. Balamurugan, "Develop New Algorithm To Improve Safety On WMSN In Health Disease Monitoring," 2022 International Mobile and Embedded Technology Conference (MECON), 2022, pp. 357-362, doi: 10.1109/MECON53876.2022.9752178.

Hadi, A. A., Sinha, U., Faika, T., Kim, T., Zeng, J., &Ryu, M. H. (2019, September). Internet of things (IoT)-enabled solar micro inverter using blockchain technology. In 2019 IEEE industry applications society annual meeting (pp. 1-5).IEEE.

Viriyasitavat, W., Anuphaptrirong, T., &Hoonsopon, D. (2019). When blockchain meets Internet of Things: Characteristics, challenges, and business opportunities. Journal of industrial information integration, 15, 21-28.

Fakhri, D., &Mutijarsa, K. (2018, October). Secure IoT communication using blockchain technology. In 2018 international symposium on electronics and smart devices (ISESD) (pp. 1-6).IEEE.

Ferrández-Pastor, F. J., Mora-Pascual, J., &Díaz-Lajara, D. (2022). Agricultural traceability model based on IoT and Blockchain: Application in industrial hemp production. Journal of Industrial Information Integration, 29, 100381.

Ferrández-Pastor, F. J., Mora-Pascual, J., &Díaz-Lajara, D. (2022). Agricultural traceability model based on IoT and Blockchain: Application in industrial hemp production. Journal of Industrial Information Integration, 29, 100381.

Raj, P., Saini, K., &Surianarayanan, C. (Eds.). (2020). Blockchain technology and applications.CRC Press.

Regnath, E., &Steinhorst, S. (2018, November). LeapChain: Efficient blockchain verification for embedded IoT. In 2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD) (pp. 1-8).ACM.

Latchoumi, T. P., Raja, K., Jyothi, Y., Balamurugan, K., & Arul, R. (2022). Mine safety and risk prediction mechanism through nanocomposite and heuristic optimization algorithm. Measurement: Sensors, 23, 100390

Mazzei, D., Baldi, G., Fantoni, G., Montelisciani, G., Pitasi, A., Ricci, L., &Rizzello, L. (2020). A BlockchainTokenizer for Industrial IOT trustless applications. Future Generation Computer Systems, 105, 432-445.

Frikha, T., Chaari, A., Chaabane, F., Cheikhrouhou, O., &Zaguia, A. (2021). Healthcare and fitness data management using the iot-based blockchain platform. Journal of Healthcare Engineering, 2021.

Bettayeb, M., Nasir, Q., &Talib, M. A. (2019, March). Firmware update attacks and security for IoT devices: Survey. In Proceedings of the ArabWIC 6th Annual International Conference Research Track (pp. 1-6).

Mohanty, S. N., Ramya, K. C., Rani, S. S., Gupta, D., Shankar, K., Lakshmanaprabu, S. K., &Khanna, A. (2020). An efficient Lightweight integrated Blockchain (ELIB) model for IoT security and privacy. Future Generation Computer Systems, 102, 1027-1037.

Hsiao, S. J., & Sung, W. T. (2021). Employing blockchain technology to strengthen security

of wireless sensor networks. IEEE Access, 9, 72326-72341.

Da Xu, L., Lu, Y., & Li, L. (2021).Embedding blockchain technology into IoT for security: A survey. IEEE Internet of Things Journal, 8(13), 10452-10473.

Medhane, D. V., Sangaiah, A. K., Hossain, M. S., Muhammad, G., & Wang, J. (2020). Blockchain-enabled distributed security framework for next-generation IoT: An edge cloud and software-defined network-integrated approach. IEEE Internet of Things Journal, 7(7), 6143-6149.

Guerrero-Sanchez, A. E., Rivas-Araiza, E. A., Gonzalez-Cordoba, J. L., Toledano-Ayala, M., &Takacs, A. (2020). Blockchain mechanism and symmetric encryption in a wireless sensor network. Sensors, 20(10), 2798.

Wang, S. Y., Hsu, Y. J., & Hsiao, S. J. (2018, December). Integrating blockchain technology for data collection and analysis in wireless sensor networks with an innovative implementation. In 2018 International Symposium on Computer, Consumer and Control (IS3C) (pp. 149-152). IEEE.