Journal of The Gujarat Research Society

ISSN: 0374-8588
Volume 22 Issue 1, January 2020

An Overview on Blockchain Tool to Manage IoT Devices

Swapnil Raj

SOEIT, Sanskriti University, Mathura, Uttar Pradesh, India

Email Id- swapnil.cse@sanskriti.edu.in

ABSTRACT: ***Blockchain technology has emerged as the next breakthrough technology since the beginning of Bitcoin in 2008. Though blockchain began as a Bitcoin core technology, it is now being applied to a wide range of fields, including banking, the Internet of Things (IoT), security, and other areas. Many business and public organisations are currently investing in technology. Aside from that, we'll see the beginnings of IoT as software and hardware develop. Those IoT devices must also be able to communicate and synchronize with one another. However, we predict that in cases where there are millions of IoT devices linked, the existing server-client approach will have some restrictions and challenges during synchronization. We can control and customize IoT devices via blockchain. Even virtually other blockchain platforms enable account as a key management system, we chose Ethereum because it allows us to administer the system in a more fine-grained manner. We employ a few IoT devices to prove an idea rather than a whole IoT system with thousands of IoT devices for the proof of concept. However, in a later study, we'd like to use blockchain to develop a fully scaled IoT system.***

KEYWORDS: Blockchain, IoT, Ethereum, Management, Smart Contract.

1. INTRODUCTION

Many people have invested in or speculated on Bitcoin since its inception in 2008 by Satoshi Nakamoto. While Bitcoin has economic, philosophical, and technical implications, none of these revolutions would be possible without blockchain, a distributed ledger that is an integral aspect of Bitcoin. Following the triumph of Bitcoin, a slew of new crypto currencies have developed, all of which are based on blockchain technology. Blockchain is being adopted in a variety of areas, not just crypto currencies. Furthermore, while Bitcoin has been a huge success, it does have some restrictions[1]. To begin with, its block generation time is around 10 minutes, which is a considerable time to complete a transaction. Second, while it can keep track of transactions using UTXOs (Unspent Transaction Outputs) and scripting, it cannot employ loops. It is not Turing complete, in other words. With these constraints in mind, Ethereum enters the picture[2] We can configure IoT devices using Ethereum. To authenticate, we can manage public key infrastructure. Ethereum can be used by IoT devices to update their behaviour. Many domains have used IoT since the era of IoT began. Attempts are being made to employ this technology in places where hundreds of devices must be connected, such as factories[3].

We can configure IoT devices using Ethereum. To authenticate, we can manage public key infrastructure. Ethereum can be used by IoT devices to update their behaviour. Many domains have begun to adopt IoT as the era of IoT has progressed. Attempts are being made to employ this technology in places where hundreds of devices must be connected, such as factories. However, there are a few drawbacks. To begin with, synchronizing all of the connected devices is a pain due to the large number of them. However, there are a few drawbacks. To begin with, synchronizing all of the connected devices is a pain due to the large number of them. As a result, we propose that Ethereum be used to manage IoT devices. We can develop a code that defines the behaviour of IoT devices using Ethereum smart contracts. We can also use smart contracts to

create public key infrastructure, preventing malevolent attackers from taking control of Ethereum's management system. We begin with a proof of concept, which consists of a few IoT devices powered by a Raspberry Pi and a smartphone. We want to use Ethereum to develop a fully scaled IoT system once we finish the model[4], [5].

1.1 Ethereum:

We can configure IoT devices using Ethereum. To authenticate, we can manage public key infrastructure. Ethereum can be used by IoT devices to update their behaviour. Many domains have begun to adopt IoT as the era of IoT has progressed[6]. Attempts are being made to employ this technology in places where hundreds of devices must be connected, such as factories. However, there are a few drawbacks. To begin with, synchronizing all of the connected devices is a pain due to the large number of them. Environment created by machines. As a result, Ethereum is one-of-a-kind in that it integrates a computing system with a blockchain. It's revolutionary because it allows programmers to write code that can run on the blockchain. Because deliberately manipulating or tampering the code will be difficult, users who rely on the written code may practically guarantee that it will operate as they expect. Even if recent assaults like as the DAO or computational denial of service occurred, they were caused by flaws in smart contract code or opcodes gas prices, not flaws in the blockchain or Ethereum itself. As a result, once the system has stabilized and matured, it will become a more powerful system.

Because the system has been stabilized, it can be used to a wide range of domains. Betting or gambling services can be built and used due to its transparency, as individuals can look at the publicly available logic or code of smart contracts. Voting services can be simply developed with the assurance that the results will not be tampered with or faked. As a result, many businesses, sectors, and individuals are attempting to develop their own Ethereum use cases.[2]

1.2 Ethereum Model:

Unlike the server-client architecture, Ethereum is a distributed computing platform, which means that all participants have access to Ethereum's blockchain Ethereum was initially described in a white paper by Vitalik Buterin, a programmer and co-founder of Bitcoin Magazine, in late 2013 with a goal of building decentralized applications. Buterin argued to the bitcoin core developers that Bitcoin and blockchain technology could benefit from other applications besides money and needed a more robust language for application development that could lead to attaching real-world assets, such as stocks and property, to the blockchain. In 2013, Buterin briefly worked with eToro CEO Yoni Asia on the Colored Coins project and drafted its white paper outlining additional use cases for blockchain technology. However, after failing to gain agreement on how the project should proceed, he proposed the development of a new platform with a more robust scripting language, a Turing-complete programming language, that would eventually become Ethereum. Despite the fact that Figure 1 resembles the server-client architecture for simplicity, the true model differs in that each contributing entity of blockchain incorporates blockchain partially or totally. Rather of transferring data to a server, each device that updates or performs transactions has Ethereum, as seen in Figure 1.

ISSN: 0374-8588
Volume 22 Issue 1, January 2020

Figure 1: Devices connected to Ethereum

Researchers understand that transactions are processed and recorded via consensus algorithm since blockchain is largely confined in contributing devices, thus attackers can't easily counterfeit or tamper with data. We can develop an IoT system that is strong enough to withstand many, if not all, denial of service and forgery assaults by leveraging this property. Despite the fact that we have only been given a few devices to work with for this paper, we believe we will be able to synchronize hundreds of them.

2. LITERATURE REVIEW

Lingjun Fan et al. in their study suggested that this paper highlights an ongoing research project that uses Ethereum Blockchain and smart contracts to create an experimental environment for IoT data management. The project's purpose is to replicate the use of IoT devices in smart city efforts and to investigate how Blockchain technology may help with IoT data management [7]. J. Dean Brocket al. in their case study suggested that the advent of the Internet of Things (IoT) brings with it new technical obstacles, such as managing a massive number of IoT devices all across the world. Despite the fact that a number of safe IoT management frameworks exist, they are all based on centralized models, which limits their application in scenarios involving a high number of IoT devices. We created a distributed IoT management solution based on blockchain to circumvent these restrictions. In this article, we compare our solution's performance to that of other IoT access management solutions[8].

Daniel Minoli et al. in their case study suggested that the Internet of Things (IoT) creates a larger attack surface, which necessitates end-to-end security mitigation. Mission-critical situations (e.g., Smart Grid, Intelligent Transportation Systems, video surveillance, e-health) to business-oriented applications are all examples of IoT applications (e.g., banking, logistics, insurance, and contract law). In the IoT, full security support is required, particularly for mission-critical applications, but also for downstream commercial applications. There have been a variety of security strategies and approaches developed and used[9]. Aafaf Ouaddah et al. in their case study suggested that In the Internet of Things, access management is a major concern. Unfortunately, current access control standards are difficult to implement on smart objects due to their restricted nature, and introducing a powerful and trusted third party to manage access control logic could

jeopardise user privacy. In this paper, we illustrate how blockchain, the promising technology that underpins Bitcoin, can be a compelling solution to the problems that arise. Fair Access is a novel decentralized pseudonymous and privacy-preserving authorization management framework that uses blockchain technology's consistency to administer access control on behalf of limited devices[10].

3. DISCUSSION

3.1. Raspberry pi:

We treat three Raspberry Pis as a metre to track power usage, an air conditioner, and a lightbulb, because utilising a genuine device like an air conditioner would need too much overhead. The policy can be put up using a smart phone. For example, when electricity usage reaches 150 KW, the user can programme equipment to go into energy saving mode. The data is transferred to the Ethereum network when the user configures the setup through smartphone. Meanwhile, gadgets like as lightbulbs and air conditioners retrieve policy values from Ethereum on a regular basis. In addition, the metre records and updates electricity usage on Ethereum. As a result, three distinct processes are taking place. Figure 2 shows formation of Raspberry pie with device.

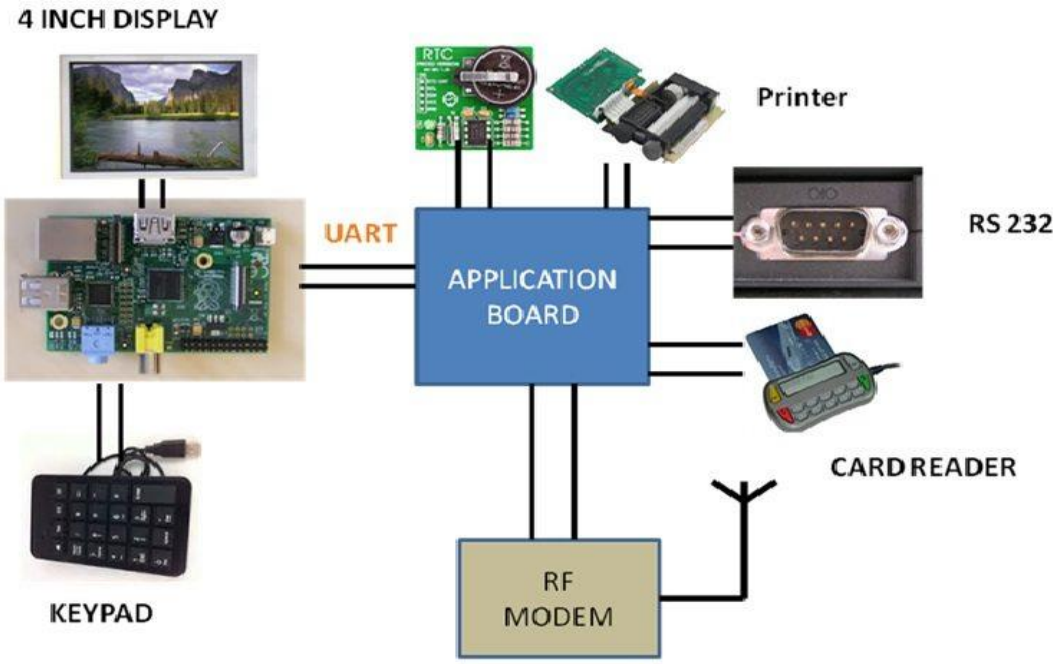

Figure 2: Formation of Raspberry pie with device.

Smart contracts are one of Ethereum's most important features. Smart contracts, which were first developed by Nick Szabo in 1994[19], provided blockchain innovation. Ethereum makes use of smart contracts, which are built on top of the blockchain.On the blockchain, programmers can write programmes. To put it another way, We can use Ethereum as a computing platform by leveraging smart contracts.platform. Solidity, for example, is a programming language.LLL and Serpent in Ethereum. Solidity is a must at this point.the most extensively used compiler and language. This level is extremely high.Once a level language has been created, it is compiled into

byte codes.Those byte codes are then used to deploy Ethereum. Since then, Byte codes are just a collection of opcodes.

3.2. Applications:

Meter Contract: On a smart contract, the metre saves electricity use on a regular basis. To be more specific, a Raspberry Pi with an Ethereum account serves as an IoT device that monitors a metre and transfers the value to Ethereum. Sender's public key and signature, as well as electricity usage, are required to prove sender's identity. The Raspberry Pi Model A and B boards have just 256 MB of random access memory in its early configurations (RAM). The early beta Model B boards defaulted to allocating 128 MB to the GPU, leaving only 128 MB for the CPU. Three distinct splits were possible on the early 256 MB Model A and B editions. The CPU was given 192 MB as a default split, which should be plenty for solo 1080p video decoding or rudimentary 3D processing. With only a 1080p framebuffer and 224 MB for Linux processing, any video or 3D was certain to fail. 128 MB was used for intensive 3D processing, as well as video decoding. The later Model B with 512 MB RAM, was released on 15 October 2012 and was initially released with new standard memory split files (arm256_start.elf, arm384_start.elf, arm496_start.elf) with 256 MB, 384 MB, and 496 MB CPU RAM, and with 256 MB, 128 MB, and 16 MB video RAM, respectively. But about one week later, the foundation released a new version of start.elf that could read a new entry in config.txt (gpu_mem=xx) and could dynamically assign an amount of RAM (from 16 to 256 MB in 8 MB steps) to the GPU, obsolescing the older method of splitting memory, and a single start.elf worked the same for 256 MB and 512 MB Raspberry Pi.

3.3. Advantages:

As all of these contracts are Ethereum-based, IoT devices like air conditioners and light bulbs must be able to retrieve values from both the metre and policy contracts.They use public key and signature to verify the validity of inputs from metre contracts. To be more specific, we're employing the RSA technique, which stores the secret keys for both the smart phone and the metre. They also use a public key and signature to verify the validity of inputs from policy contracts. When electricity consumption exceeds policy while retrieving values on a regular basis, devices automatically convert to energy-saving mode. Although Ethereum accounts can be used as public keys, it is not recommended. The majority of Raspberry Pi systems-on-chip can be overclocked to 800 MHz, and some can even be overclocked to 1000 MHz. According to sources, the Raspberry Pi 2 can be overclocked to 1500 MHz in extreme instances (discarding all safety features and over-voltage limitations). The overclocking choices on boot in the Raspbian Linux distro can be done with a software programme called "sudo raspi-config" without voiding the warranty. In those cases, the Pi will turn off overclocking if the chip temperature reaches 85 degrees Celsius (185 degrees Fahrenheit), but it is possible to override automatic over-voltage and overclocking settings (voiding the warranty); an appropriately sized heat sink is required to protect the chip from serrated edges. Newer versions of the firmware provide the option to select from five overclock ("turbo") presets that, when enabled, seek to maximise the SoC's performance while preserving the board's lifetime. This is accomplished by continuously monitoring the chip's core temperature and CPU load, as well as dynamically altering clock rates and core voltage. Performance is throttled when demand on the CPU is low or it is running too hot, but if the CPU is busy and the chip's temperature is appropriate, performance is momentarily

enhanced with clock speeds of up to 1 GHz, depending on the board version and which of the turbo settings is employed.

3.4. Working:

We used Ethereum to deploy smart contracts. Following the deployment of contracts, we began providing inputs after encoding.Once the settings have been successfully updated/registered on We were successful in retrieving values from Ethereum.The following is a picture of our completed prototype. Figure 3 shows the Chips Set Of Resberrypie Model.

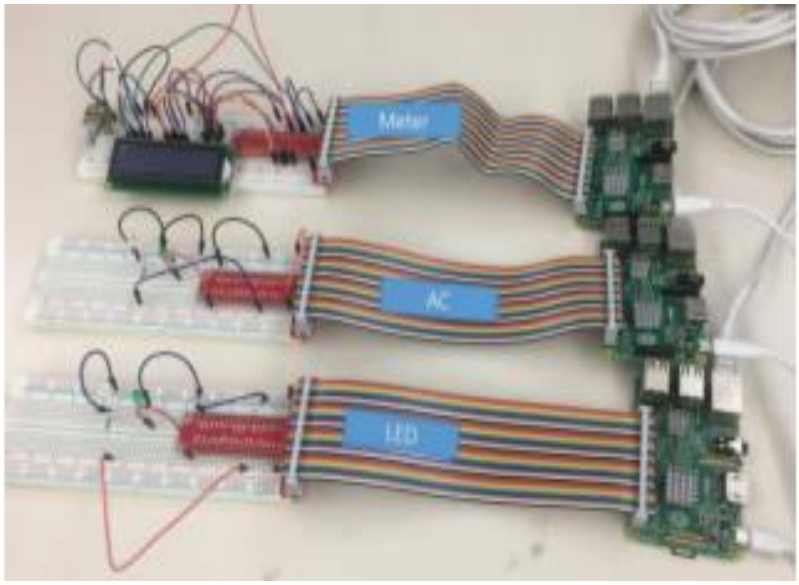

Figure 3: Chips Set Of Resberrypie Model

During the development process, we discovered various flaws in the Ethereum blockchain. To begin with, even though it has a transaction time of roughly 12 seconds, it is still too slow for some sites. It may be challenging to use such technologies in time-sensitive domains. Second, since the light client isn't currently supported on Ethereum, we'll either need to utilize a proxy or have a lot of storage to save the complete blockchain. Using a proxy server may be simple. However, because there is a third party involved, we end up jeopardising security. While the second option does not compromise security, it may necessitate massive storage, which would be prohibitively costly or infeasible.

4. CONCLUSION

In this study, we propose using Ethereum, a blockchain computing platform, to control IoT devices. We compose smart contracts to save data from meters and smart contracts to store data from smart meters phone. Using an Ethereum account, the metre provides data on a regular basis. The usage of power and the use of a smart phone transmits policies for air pollution. We're starting with a modest number of devices as a proof of concept. Since we discovered that such a system can be built, we would aim to develop a fully-scaled IoT system with a large number of devices in future research. With the commencement of this project, we hope to witness advancements in IoT, where customers don't have to worry about synchronization or denial of

service assaults while being served efficiently and quickly. The Raspberry Pi is a credit-card size computer developed in 2012 at the University of Cambridge's Computer Laboratory, The Pi costs only $35, runs Linux in a graphical environment, and provide GPIO (general purpose I/O) connectors for sensors and motors. In this tutorial we will introduce the Pi and provide a guide to using it, include hardware and software requirements. We will also describe how we are using the Pi in undergraduate computer offering ranging from introductory program courses to upper-level systems offerings.

REFERENCES

[1] J. Brito and A. Castillo, "Bitcoin: A Primer for Policymakers," *Mercat. Cent. Geroge Mason Univ.*, 2013.

[2] W. G. Ethereum, "A secure decentralised generalised transaction ledger [J]," *Ethereum Proj. yellow Pap.*, vol. 151, pp. 1–32, 2014.

[3] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash SyNakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Consulted, 1–9. doi:10.1007/s10838-008-9062-0stem," *Consulted*, 2008.

[4] J. Bughin, M. Chui, and J. Manyika, "An executive's guide to the Internet of Things," *McKinsey Quarterly*. 2015.

[5] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Networks*, 2010.

[6] M. Hartwig, "ECDSA Security in Bitcoin and Ethereum : a Research Survey," *Blog.Coinfabrik*, pp. 1–10, 2016.

[7] L. Fan, G. B. Burke, J. R. Gil-Garcia, X. Hong, and D. Werthmuller, "Investigating Blockchain as a data management tool for IoT devices in smart city initiatives," in *ACM International Conference Proceeding Series*, 2018.

[8] J. D. Brock, R. F. Bruce, and M. E. Cameron, "Changing the world with a Raspberry Pi," *J. Comput. Sci. Coll.*, 2013.

[9] D. Minoli and B. Occhiogrosso, "Blockchain mechanisms for IoT security," *Internet of Things*, 2018.

[10] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "Towards a novel privacy-preserving access control model based on blockchain technology in IoT," in *Advances in Intelligent Systems and Computing*, 2017.