**SECURING PRIVATE MESSAGING USING TRUSTLESS 2FA SMART CONTRACT IN BLOCKCHAIN**

***PART ONE – PROJECT PLAN***

**INTRODUCTION**

Blockchain is a new technology that has the potential to significantly increase transaction security in banking, supply chain, and other networks. It effectively serves as the foundation for a dynamic distributed ledger that can be used to minimise time when capturing transactions amongst users, eliminate intermediary costs, and reduce fraudulent and meddling risks.

As mentioned in the literature review below, blockchain has a lot of applications and one of the major and trending application that I will be focusing of those applications is ‘how blockchain is applied to make cybersecurity’ better. Despite the fact that blockchain was first proposed as a basic technology for Bitcoin. The fundamental principles of blockchain are disclosure, openness, and exchange. Instead of relying on a third-party middleman, blockchain ensures trust by employing cryptographic and statistical algorithms. Blockchain is starting to become the technical heart of cryptocurrency, asset management, credit control, and other industries by effectively guaranteeing the validity and originality of transactions. In a nutshell, the technical and security advantages allowed plenty of room for blockchain development. Furthermore, blockchain's revolutionary privacy and trust mechanism has opened up the possibility of using it to solve certain critical cybersecurity issues.

The following are some examples of blockchain application scenarios in cybersecurity (Dai, 2017):

* Decentralized Distributed Secure Domain Name Service
* Keyless Signature Infrastructure
* Secure Data Storage
* Trustless 2 Factor Authentication etc.,

Two-factor authentication (2FA) is commonly used in banking, email, and virtual private network (VPN) connections, as well as in gaining access to any secure web service (Cristofaro, 2014). In 2FA, users are required to submit additional secret information in addition to their password in order to be authenticated. Upon receiving an authentication request from users, a centralised trusted third party often generates this secret information (tokens). As a result, this extra layer of protection comes at the cost of implicitly trusting the third party's services. The security of such authentication systems is constantly at risk of being hacked by the trusted party. By adding an additional layer of security component to the OpenSSH server, a commonly used application for Secure Shell (SSH) protocol, the present solution demonstrates the idea of employing blockchain technology for 2FA. (Amrutiya el at., 2019)

**AIM OF THE PROJECT**

As mentioned in the title already, my project will focus on making an attempt to tune the Trustless 2FA using Smart Contract using Blockchain so that private messaging is more secure. This is an experimental attempt that is being made. The certainty about a successful and positive result is also experimental.

**OBJECTIVES**

* Objective 1: Comprehensive study about blockchain, application of blockchain in cybersecurity, Two Factor Authentication, Existing Trustless 2FA solution
* Objective 2: Establishing the brief idea about how the solution will be implemented.
* Objective 3: Gathering all the resources required and learning how to use and implement the same.
* Objective 4: Setting up the resources.
* Objective 5: Testing the implementation and determine the percent of success achieved.
* Objective 6: Report and presentation preparation.

**TASKS**

**Phase 1**: Accomplishing Objective 1

* Task 1A: Go through 20+ academic, conference or journal papers for first phase of literature review.
* Task 1B: Prepare the draft of the first phase of literature review.
* Milestone 1A: Submit the Literature review in the Dropbox.
* Task 1C: Connect with supervisor for formative feedback to continue with the research.
* Task 1D: Update Literature review with more references and append the Project Plan.
* Milestone 1B: Submit the Research Method Assessment.

**Phase 2**: Accomplishing Objective 2

* Task 2A: Go through academic, conference, journal papers and videos to understand what technologies can be used.
* Task 2B: Connect with supervisor for suggestions about technologies to use.
* Milestone 2: Make a list of most of the finalized technologies to be used.

**Phase 3**: Accomplishing Objective 3

* Task 3A: Go through academic, conference, journal papers and videos to understand how the resources needs to be used.
* Task 3B: List all the sources from where the resources will be utilized.
* Task 3C: Connect with supervisor and get the resource list reviewed.
* Milestone 3: Implement/install all the resources needed.

**Phase 4**: Accomplishing Objective 4

* Task 4A: Set up the implemented/installed resources by modifying according to the solution needed.
* Task 4B: Connect with supervisor to cross check if the resources are set up correctly
* Milestone 5: The resource set up is completed and ready for testing.

**Phase 5**: Accomplishing Objective 5

* Task 5A: Go through academic, conference, journal papers and videos to understand how and to what extent testing needs to be done.
* Task 5B: Seek guidance from supervisor about how the testing needs to be performed.
* Task 5C: Make note of every outcome based on the testing performed.
* Milestone 5: Testing phase is finished with results in hand.

**Phase 6**: Accomplishing Objective 6

* Task 6A: The documentation will be an ongoing process.
* Task 6B: Refer the student handbook and research method module class to understand how the report needs to be prepared.
* Task 6C: Get the documentation reviewed by supervisor after every significant section/change Is drafted.
* Task 6D: Make changes to the report based on formative feedback from supervisor.
* Task 6E: Complete the final draft of the project report with the results and findings and share it with supervisor for final review.
* Task 6F: Prepare presentation slides (create video demo if required and append in the presentation slide)
* Milestone 6: Finish the presentation and submit the report to Dropbox

**RESOURCES**

**Blockchain**

* Ethereum

Ethereum is a blockchain with a Turing-complete computer program built in. It has an abstract layer that allows anyone to define their own ownership, transaction formats, and state transition algorithms. This is accomplished through the use of smart contracts, which are a collection of cryptographic rules that are only performed if specific criteria are met. (Wood, 2017)

* Solidity

Every node in the Ethereum network is controlled by EVM, which executes its commands. Smart contracts are converted into EVM code, which is subsequently executed by nodes. (Wood, 2017) Solidity is one of the most widely used programming languages for creating smart contracts. In this project, I will be using Solidity to create Ethereum smart contract. Solidity is like a java script programming language. Contracts are structured similarly to classes in object-oriented programming languages when utilising Solidity for contract development. Contract code, like traditional imperative programming, comprises of variables and functions that read and alter them. (Maximilian Wohrer el at., 2018)

* Truffle Framework

Truffle Framework is a must-have tool for Ethereum smart contract developers. It creates smart contracts using the Solidity programming language. It also provides a framework for testing Smart Contracts and the tools needed to conduct transactions (deploy) Smart contracts on the blockchain network. (Ali Mansour Al-madani el at., 2020) (Bhosale, 2019)

* Ganache – cli

As this project is on a personal scale rather than commercial level. I will be using a tool called Ganache. Ganache is a tool for deploying smart contracts and testing them. Ganache gives ten 100Ether accounts for testing our smart contracts on local blockchains. (Bhosale, 2019) (Ali Mansour Al-madani el at., 2020)

* Metamask

Metamask is an extension that is used to connect to the Ethereum. It is available as the Google extension. MetaMask now connects to the local Ethereum network, which is used for this application, and provides a personal account as well as the ability to interact with the smart contract application that was built before this stage. (Ali Mansour Al-madani el at., 2020) (Bhosale, 2019)

**Website/Device Application**

After looking up at options using GitHub (GitHub, n.d.), either Web application or an application will be installed on the device. This application will act as the client and server side interface where the blockchain will be implemented and tested. Further research is required to decide what kind of application will be set up.

**Programming Language**

Based on what type of application will be implemented, additional programming language will need to be used. However, currently Java script like React.js (front end) and Node.js (back end) will be used.

**PROJECT RISK ASSESSMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl No** | **RISK** | **EFFECT** | **MITIGATION** |
| 1 | Loss or damage to laptop | Progress of the project may be stopped for a while | Backup files regularly |
| 2 | Vulnerability in the application | May be threatening to the entire system | Need to do a genuine check. |
| 3 | Large file size | May tamper the performance of the laptop | Make sure enough space is there on the SSD and External Hard Drive |
| 4 | Slow internet | May stall the project a little bit | Make sure to have enough browsing data |
| 5 | Improper installation of resources | May hamper the performance of the resources and its implementation | Perform thorough research and then instal the resources |

**PROFESSIONAL ISSUES**

* Legal Issues

The project is carried out in front of the general public, in accordance with the BCS code of conduct. In the data set used, there was no information about people's privacy or security. The GDPR (General Data Protection Regulation) is not infringed upon in any way.

* Social Issues

Blockchain and cyber security may be considered as the next generation's interconnectivity paradigm, allowing communication between gadgets and machines and allowing activities to take place without the need for human intervention, but they are also prone to cyber-attacks and viruses. The main goal of this project is to provide a secure private messaging system for users that prevents hackers from stealing OTPs.

* Ethical Issues

There will be no ethical issue as no third party data will be used. Only my credentials will be used and so will the OTP be used only by me. The project's goals and objectives have no bearing on ethical issues. All sources used in the literature review, as well as the data obtained from them, shall be adequately mentioned and cited.

**TIMELINE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Task** | **Time to Complete** | **Resources** | **Comments** |
| July 15 | Deploy either web/device application | 2 hours | Laptop + Software + Internet |  |
| July 31 | Install all the required resources | Half a day | Laptop + software + Internet |  |
| August 15 | Modify all the resources according to the project requirement |  | Laptop + software + Internet | On-going progress expected to finish by August 15 |
| August 30 | Finish testing of the solution implemented |  | Laptop + software + Internet | On-going progress |
| September 15 | Finish preparing the report and presentation |  | Laptop + software + Internet | On-going progress |

**GANTT CHART**



**Figure 1: Gantt Chart of the proposed project** (Ramakrishnan, n.d.)

***PART TWO – LITERATURE SURVEY***

**INTRODUCTION AND INVENTION OF BLOCKCHAIN**

The 21st century is considered to be the evolution of digitalization worldwide. From hardware like a floppy disk to nanorobots and software like operating systems to applications supporting day to day functionality, technology has become a crucial part of every individual. Considering the requirements of humans' which will make the day-to-day functionality a cakewalk, multiple inventions and advancements are fabricated. Today's digital world is built to take care of essentials like online shopping for food, clothing, and shelter to requirements like social media platforms to communicate with people worldwide to network formally and casually to entertainment to save lives of animals and human beings.

Once the basic requirements of humans' were addressed, the inventions then began to fabricate around gadgets and software to perform fancy tasks keeping luxury in mind.

Timeline

Description automatically generated

**Figure 1: Evolution of Digitalization and Technology** (Toldo, 2018)

Amidst all these frenzy inventions, in 2008, Satoshi Nakamoto proposed the need for an entirely new financial system compatible with the digital world as it would not be safe enough to rely on a third-party financial system altogether due to possible security flaws either in the software or hardware. This thought process would be one of the most potent whitepaper written for both the economics world and the digital world. He stated that "Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the System works well enough for most transactions, it still suffers from the inherent weaknesses of the trust-based model." (Nakamoto, 2008)

Just like how digitalization paved the way for secured communication between sources by utilizing encrypting methodology, cryptography. (Zhai, 2019) An electronic payment system that did not depend on trust and instead used cryptographic proof between two parties had to be built. Such payment method almost after a decade later came to be known as Bitcoin, the cryptocurrency. The methodology behind cryptocurrency was that "a double-spending problem using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions. The System is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes." (Nakamoto, 2008)

Cryptocurrency is also known as the electronic coin. The first party passes the e-coin to the next party after signing a hash of the past transaction digitally and appending the next recipient's public key, giving the flexibility to the first party to verify the series of ownership verifying the signatures.

As the transaction is being performed in stages, a server with a timestamp is used. A timestamp server works by taking a hash of a block of items to be timestamped and widely publishing the hash, such as in a newspaper. (Nakamoto, 2008) The timestamp proves that the data had to exist at the time to be included in the hash. Every individual time stamp is considered as a Block. These blocks containing the preceding timestamp form a chain, with each subsequent timestamp reinforcing the previous ones. The use of this methodology paved the way for the invention of a new technology known as Blockchain.

Diagram

Description automatically generated

**Figure 2: Blockchain Structure** (Joshi, 2018)

**EVOLUTION OF BLOCKCHAIN**

Over the six years, though Blockchain was primarily created to establish the peer-to-peer financial System electronically, people have utilized the workflow of Blockchain in multiple usage aspects. While businesses have been sceptical about adopting Blockchain actively, many have begun testing the technology and integrating it at a minimal concentration to test how it impacts the productivity of the organization. The experimentation of using Blockchain since 2019 has rapidly increased too. A blockchain is a distributed database that stores data in a decentralized network. Christian Cachin mentions the four elements duplicated elements as the ledger, cryptography, consensus and business logic. (Cachin, 2017) According to International Data Corp, total corporate and government spending on the Blockchain should hit $2.9 billion in 2019, an increase of 89% over the previous year, and reach $12.4 billion by 2022. When PwC surveyed 600 execs last year, 84% said their companies are involved with Blockchain. (Castillo, 2019)

The following are the application where Blockchain is being implemented currently:

* Cryptocurrencies
* Smart Contracts (Bernhard Waltl el at., 2017) like Ethereum (Buterin, 2014) and Hyperledger
* Financial Services like Bitcoin (Nakamoto, 2008) and Ripple (Kang, 2016)
* Gaming
* Supply Chain Management like mining, food etc.,
* Healthcare
* Ledger System for Sales, Digital Payments etc.,
* Used for mitigation of attacks in cybersecurity

Chart

Description automatically generated

**Figure 3: Uses of Blockchain** (Martel, 2018)

Of all the possible uses of Blockchain, one of the crucial areas where the experimentation and implementation of Blockchain are rapidly growing is in the domain of cybersecurity. Anything too good can also turn out to be wrong, which is what is happening in the virtual world. With most people depending on the internet, the world wide web has become the most extensive repository for all sorts of data and information; this has grabbed the attention of the bad guys who try and steal data for personal revenge, benefit, financial grip or even to exert power. The number of cybercrimes is exponentially increasing with each day passing. As per the Security Intelligence Report, the average cost of a cyber-attack data breach of 2019 was $3.92 million. (Legrand, 2020) On the alternative, the expense of hacking is minimal, as most of the cyber-attack software resource is available for free or as low as one dollar on the Dark Web. The idea of needing an average of 5 minutes to hack an IoT system becomes even more troubling. Hence having the best defence against these kinds of cyberattacks has become essential.

**THE USES OF BLOCKCHAIN IN CYBERSECURITY**

As designed and intended, the technology is credited for its information integrity assurance. One of the best uses would be utilizing its integrity assurance for building cybersecurity solutions for many other technologies. (Legrand, 2020)

A few of the use cases of cybersecurity where Blockchain can be used are:

* Data Transmission Protection
* Cyber-Physical Infra Verification
* The origin of Secure Computer Software
* Preventing and Securing DDoS and DNS
* Medium Storage Decentralization
* IoT Security
* Private Message Securing

Logo, company name

Description automatically generated

**Figure 4: Use Cases of Cyber Security Where Blockchain Can Be Implemented** (Mire, 2018)

**IMPORTANCE OF 2 FACTOR AUTHENTICATION**

Today, the internet is the colossal plug that keeps the entire world connected and bonded by being the standard medium for communication, that too when the world is on hibernate mode due to the Covid-19 Pandemic. With so many different kinds of cyberattacks, there is a constant need for upgrading security when it comes to communication. Therefore, implementing and using Blockchain to secure private messaging between peers needs to be given utmost attention. There are many ways to secure private messaging, and one such method is the use of 2 Factor Authentication (2FA). However, hackers even try to look for and exploit loopholes in 2FA. The loophole that exists in 2FA is relatively less compared to others as such. Hence, by using Blockchain Smart Contracts, trustless 2FA can be created. It can be achieved by adding an extra layer of security component to the OpenSSH server, a widely used application for Secure Shell (SSH) protocol. (Amrutiya el at., 2019)

**IMPLEMENTATION OF TRUSTLESS 2FA**

According to Amrutiya (2019), using Blockchain Smart Contract, primarily the need for centralized third-party token providers can be eliminated. Secondly, the potential logic behind the program that will generate tokens and distribute the same to the users in a decentralized manner can be created by programming the smart contracts using Ethereum. The proposed solution is built on the Ethereum blockchain and then incorporated with the OpenSSH server to enable OpenSSH clients to use two-factor authentication. The authentication process is implemented using the following steps:

* Ethereum:

Ethereum is a secure decentralized ledger that offers a scalable and user-friendly ecosystem for building trustless, decentralized applications. (Buterin, 2014) Ethereum is a distributed computing platform that allows anyone to execute intelligent contracts in return for a fee. The fairness of the smart contract's execution relies on the miners and the underlying consensus protocol, which assures that certain ethical miners are credited.

* Smart Contract

Ethereum offers smart contracts, which can be engineered to execute application logic depend on a range of process parameters or events, as well as using the decentralized ledger's non-volatile storage. Smart contracts are mainly programmed in a language called Solidity which upon compilation is converted into a bytecode that can be executed on the Ethereum Virtual Machine. (Mukhopadhyay, 2018)

* Decentralized Application

A decentralized application (DApp) (Mukhopadhyay, 2018) is a service that allows users to communicate with Smart Contracts in a user-friendly way. To connect with the Ethereum blockchain, Ethereum DApps usually use an HTML/JavaScript web application and a JavaScript API. DApp connects to local blockchain node via JSON-RPC API (Shin, 2014)

* Pluggable Authentication Modules

A pluggable authentication module provides a Linux operating system to integrate multiple low-level authentication schemes into a high-level API. (Samar, 1996)

* Software Stack

Diagram

Description automatically generated

**Figure 5: Blockchain-Based Software Stack for 2FA** (Amrutiya el at., 2019)

**WORKFLOW OF TRUSTLESS 2FA FROM USER PERSPECTIVE**

**Step 1**: Employees sign up for a framework that is part of this infrastructure.

**Step 2**: Staff must also enrol the users who will be capable of connecting with the smart contract.

**Step 3**: The user views the entire list of available systems and sends a request to the system that he intends to use.

**Step 4**: All the requests sent from the user are reviewed and accepted by the staff.

**Step 5**: For the authorized System, clients can obtain an OTP and save it in Blockchain.

**WORKFLOW OF TRUSTLESS 2FA (BACK END)**

**Step 1**: The user establishes an SSH link to the device he wants to use.

**Step 2**: As part of the standard SSH authentication process, SSH will initially authenticate the user with a username and password.

**Step 3**: The PAM module uses JSON RPC API calls to query the Blockchain to validate if the System has been repealed. If the device has already been marked as revoked, the user will be denied entry.

**Step 4**: The user will be prompted to enter his OTP to gain entry if the System is not revoked. At this stage, the module will collect all the tokens that are stored as SHA3 hashes.

**Step 5**: A request is generated to delete the token that matches the SHA3 hash (OTP) entered by the user. The user would then be provided access to the System while preventing the reuse of the same token in future.

**Step 6**: If the OTP is inaccurate, an error message will be shown accordingly, and after three incorrect OTP entries, the device will be marked as revoked in the Blockchain, and no further ssh access will be granted before the System is de-revoked by workers.

However, the above mentioned steps of implementing the Trustless 2FA is from the view-point and understanding of the workflow that is already in place.

The hopeful attempt that is going to be made by me is to try and implement a Trustless 2FA system using Smart Contract using Blockchain. This methodology will not be using OpenSSH, instead the approach is to detach the smart contract from the devices/web application where the OTP authentication is required. Rather than installing Ethereum clients on each server, we can try to create a solution that authenticates OTPs alongside the users’ blockchain private keys such that the OTP and the private key will be inter-linked; where when the user tries to authenticate, he will need to digitally sign and the signature will be verified by the smart contract. (Ajienka, 2021)

Diagram

Description automatically generated

**Figure 6: Tentative Workflow of Trustless 2FA Using Blockchain** (Amrutiya el at., 2019)

**LIMITATIONS**

While the proposed workflow does not require the integration of SMS or mailing system since the user generates the random token himself, which is in contrast to the conventional design, there are a couple of drawbacks according to Amrutiya (2019)

* Any modifications to the smart contract will necessitate changes to all PAM modules installed on the servers.
* Since nodes depend on each other to pull the latest changes in the smart contract, network downtime will obstruct ssh access.
* All PAM modules will need access to Ethereum private keys to be able to digitally sign JSON RPC API calls for such a Popular gateway architecture, and such a library is not available in C as of now.

**CONCLUSION**

Based on the literature survey, the primary factor that has been understood is that a workflow and implementation of a Trustless 2FA using smart contract Blockchain is already in place but with certain limitations. Hence, the focus hereafter will be to fine-tune and overcome the limitations mentioned above.

The board idea currently for fine-tuning will be based on researching and experimenting to answer the below questions:

* Rather than installing Ethereum clients on each server, we can try to create a solution that authenticates OTPs alongside the users’ blockchain private keys such that the OTP and the private key will be inter-linked; where when the user tries to authenticate, he will need to digitally sign and the signature will be verified by the smart contract. (Ajienka, 2021) Is this possible?
* What programming language(s) will be compatible to support the popular gateway architecture?
* Can the OTPs be generated in a decentralised manner instead of relying on the existing centralised authentication methodologies? (Ajienka, 2021)

In submitting this work, I confirm that I am aware of, and am abiding by, the University's expectations for proofreading.

**REFERENCES**

1. Ajienka, N., 2021. Microsoft Teams Meeting - Project Supervision Call. Nottingham: s.n.
2. Ali Mansour Al-madani el at., 2020. Decentralized E-voting system based on Smart Contract by using Blockchain Technology. s.l., 2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC).
3. Amrutiya el at., 2019. Trustless Two-Factor Authentication Using Smart Contracts in Blockchains. s.l., 2019 International Conference on Information Networking (ICOIN).
4. Bernhard Waltl el at., 2017. Life Cycle of Smart Contracts in Blockchain Ecosystems. Datenschutz und Datensicherheit - DuD, 41(8), pp. 497-500.
5. Bhosale, K. e. a., 2019. Blockchain based Secure Data Storage. International Research Journal of Engineering and Technology, 6(3), p. 4.
6. Buterin, V., 2014. A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM. White Paper, 3(37), p. 36.
7. Castillo, M. d., 2019. Forbes. [Online]   
   Available at: https://www.forbes.com/sites/michaeldelcastillo/2019/04/16/blockchain-50-billion-dollar-babies/  
   [Accessed May 2021].
8. Cachin, C., 2017. Blockchain, Cryptography, and Consensus: Invited Presentation at the 10th Interaction and Concurrency Experience. Electronic Proceedings in Theoretical Computer Science, Volume 261, pp. 1-1.
9. Cristofaro, E. D. e. a., 2014. A Comparative Usability Study of Two-Factor Authentication. s.l., USEC 2014.
10. Dai, F., 2017. From Bitcoin to cybersecurity: A comparative study of blockchain application and security issues. s.l., 2017 4th International Conference on Systems and Informatics (ICSAI).
11. GitHub, n.d. GitHub: Where the world builds software. [Online]   
    Available at: <https://github.com/> [Accessed June 2021].
12. Joshi, A., 2018. Research Gate. [Online]   
    Available at: <https://www.researchgate.net/figure/Blockchain-Structure_fig1_325173502> [Accessed May 2021].
13. Kang, A. e. a., 2016. UNDERSTANDING AND REGULATING TWENTY-FIRST CENTURY PAYMENT SYSTEMS: THE RIPPLE CASE STUDY. Michigan Law Review, 114(4), pp. 649-681.
14. Legrand, J., 2020. Cyber Management Alliance. [Online]   
    Available at: https://www.cm-alliance.com/cybersecurity-blog/the-future-use-cases-of-blockchain-for-cybersecurity [Accessed May 2021].
15. Martel, K., 2018. United States Cybersecurity Magazine. [Online]   
    Available at: https://www.uscybersecurity.net/csmag/blockchain-good-bad-ugly [Accessed May 2021].
16. Maximilian Wohrer el at., 2018. Smart contracts: security patterns in the ethereum ecosystem and solidity. s.l., 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE).
17. Mire, S., 2018. Disruptor Daily. [Online]   
    Available at: <https://www.disruptordaily.com/blockchain-use-cases-cyber-security/> [Accessed May 2021].
18. Mukhopadhyay, M., 2018. Ethereum Smart Contract Development. In: P. P. Ltd, ed. Ethereum Smart Contract Development: Build blockchain-based decentralized applications using solidity. Birmingham : Packt Publishing Ltd, pp. 103-144.
19. Mukhopadhyay, M., 2018. Ethereum Smart Contract Development. In: P. P. Ltd, ed. Ethereum Smart Contract Development: Build blockchain-based decentralized applications using solidity. Birmingham: Packt Publishing Ltd, pp. 145-164.
20. Nakamoto, S., 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. [Online]   
    Available at: https://bitcoin.org/en/bitcoin-paper [Accessed May 2021].
21. Ramakrishnan, S., n.d. teamgantt.com. [Online]   
    Available at: https://www.teamgantt.com/  
    [Accessed June 2021].
22. Samar, V., 1996. Unified login with pluggable authentication modules (PAM). New York, Association for Computing Machinery.
23. Shin, S. e. a., 2014. Introduction to JSON (JavaScript Object Notation). http://www.cse.iitd.ac.in/ cs5090250/JSON.pdf, p. 30.
24. Toldo, L., 2018. Research Gate. [Online]   
    Available at: https://www.researchgate.net/figure/Evolution-of-Information-Technologies-IT-from-digital-era-into-the-intelligence-era\_fig1\_329624902  
    [Accessed May 2021].
25. Wood, G., 2017. ETHEREUM: A SECURE DECENTRALISED GENERALISED TRANSACTION LEDGER. Yellow Paper, Volume 150, p. 34.
26. Zhai, S. e. a., 2019. Research on the Application of Cryptography on the Blockchain. Journal of Physics: Conference Series, Volume 1168, p. 032077.