Immigration Control and Management System using Blockchain

Abhay Goel[1], Abhishek Sharma[2], Dr Deepak Gupta[3]
Department of Computer Science and Engineering[1,2,3]
Maharaja Agrasen Institute of Technology[1,2,3]
abhaygoel1906@gmail.com[1], abhishekcrdsa7@gmail.com[2], deepakgupta@gmail.com[3]

Abstract— In this paper, we propose a system using Blockchain technology to create a decentralized, secure, and scalable migration records of individuals. We try to utilize Ethereum Blockchain and Proof of Work as the consensus algorithm. We attempt to mitigate the process of illegal immigration by keeping immutable and unique record of state of migration as well as personal information of an individual to check their authenticity. The AWS S3 service has been used to store the official documents of each individual securely, while still maintaining transparency. The proposed system will not only help to check for unlawful immigration but will also allow to check if an individual has successfully reached the intended destination. Also, the storage system will help keep the original documents of every individual in a cloud based storage solution.

Index Terms— migration, blockchain, hyperledger, immutable, smart contracts, AWS S3.

I. INTRODUCTION

When individuals leave their hometown to deport to another nation, this act of movement is called migration. These individuals adopt the process of immigration to become the permanent residents of the new nation. Generally, the traditional procedure of immigration is very tedious and repetitive. The applicant needs to get a valid Visa [1] and then apply for permanent residency which later becomes the nation's citizenship. This process is eased if the applicant is backed by a company or if a family member of the applicant is already the resident of the nation. Due to the immigration laws of some countries being very strict, People take illegal practices and infiltrate the country without permission. This leads to unofficial migration of many.

This paper aims at solving such problems by introducing a ledger system that can record each immigration and emigration in the country, by utilizing the advantages of Blockchain [2] technology implemented by using Ethereum [3]. Any person who is not a native or in the records would be classified as an illegal immigrant. The system would be decentralized as the Blockchain stores its data at various nodes in the world. The data stored in this system would be immutable [4] as it is impossible to make changes to the data present in Blockchain. The AWS S3 service has been used to store the official documents of each individual on the cloud. This would lead to various files being stored at a reliable destination which are easily retrievable when necessary (as their stored over the cloud). This data can only be retrieved by the administrator of the application. Moreover, it would be seamless as the data entered in the system would have high availability due to no single point of failure and data redundancy at every node.

In this paper, we have deployed a system that stores the information of all the individuals that are leaving their country to reside in a new one, even if temporarily. The records stored in the system can be present at more than one locations at the same time [5]. The system would be synchronous at all the ports in which it is present. The official documents of the migrants, such as the passport and other valid and legal documents assigned to the individual by the government, would be stored over the cloud. These would be accessible to all the ports at the time of their storing therefore, enhancing the system's synchronization and decentralization.

II. PROPOSED WORK

The problem of illegal immigration has always been an issue faced by almost every other country in the world. In a nutshell, the proposed work keeps a track of immigrants by storing several relevant pieces of information in the form of immutable [6] and unique records in Blockchain. When a person in suspected of illegal immigration, his official documents are matched against the record present in Blockchain and the legal documents uploaded on the cloud. The process can be of migration can be basically divided into the following phases.

A. Emigration

At the time of emigration [7], a new record is stored in the Blockchain by making a transaction. The transaction contains necessary information of the user that will be matched with the individual's official documents when he is suspected of unlawful migration. These official documents can be retrieved from the AWS S3 cloud services at any time by the authorized personnel. These records would be useful when the governments of any country would need to check for the details of the respective individuals as mentioned in their legal records. Along with this, the transaction also contains the state of process of immigration such as information about source, destination and overall process of immigration. This record is helpful while verifying the legitimacy of the immigration process undergone by the respected person while entering the country at the other side.

B. Immigration

When an individual legally enters into another country, the state of migration [8] is marked as complete. The details of

individual are checked and confirmed with those present in the legal documents of the individual.

Moreover, the information about the individual reaching the intended destination is also marked as true. A transaction is made to update all the relevant information. Here, the transaction has been completed successfully as both the immigration and the emigration credentials of the individual are present.

C. At the time of checking

When a check is done, an individual's official documents are matched against the record in Blockchain. These legal documents usually contain (but not restricted to) the soft copy of the official passport of the individual that has been stored on the cloud. The fields that stores the overall state of the migration process as well as the fields that gives information about if the correct procedure was followed during emigration cumulatively tells if the person is a legal immigrant or not.

The next section discusses about the method which would be followed to bring the application to work. The method would be demonstrated in the form of a flowchart to explain the working of the software in detail. The flowchart is demonstrated twice for better explanation of the various elements that go in the working of the application.

III. FLOWCHART

The following demonstrates the method to be followed. Whenever an individual is suspected for an illegal immigration, this process can be followed. It not only eliminates the hassle for the individual who has been suspected but also for the person who has to check for the illegal immigration. The process is very simple as it only requires to match the user's information with the personal documents.
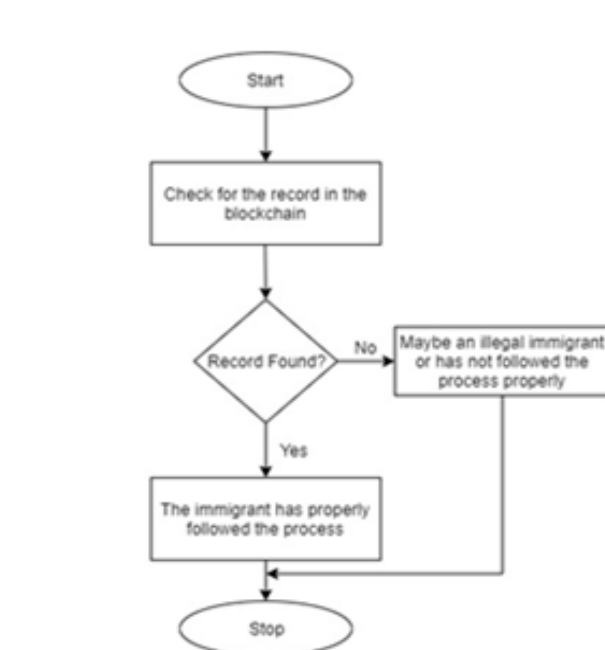

Fig. 1. Flowchart to check for illegal immigration.

The following flowchart depicts the process and the basic information that needs to be stored during the process of immigration. A transaction is done at the of start migration process which stores the information of the individual with the details about the source and destination. It also stores the state of the migration process. When the migration complete, a transaction is again done to update the information of the individual and marks the process of migration as complete.
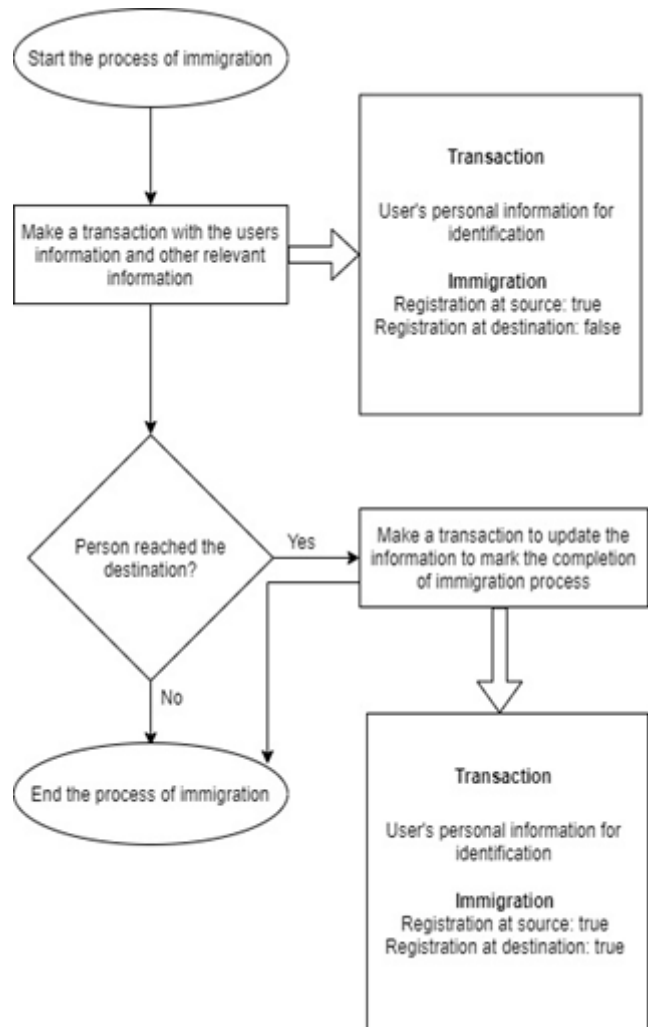

Fig. 2. Flowchart demonstrating the process to be followed at the time of migration.

IV. CASE STUDY

To explain the flowchart from the previous section, here is a case study of the software. A web application has been developed and the smart contract [9] that governs the interaction of the application with the Blockchain, has been deployed on the Rinkeby test network. The web application is connected to the wallet of the user hence if transactions are made, money will be deducted from the user's wallet. As mentioned, and the application is deployed to a test network so it would receive money from the test wallet.

The passport and other legal (official) files that are needed to be uploaded to the server by the individual, would be uploaded to the AWS S3 bucket which has been created for this instance.

The application contains all the necessary functionalities and the respective security concerns are met by using the various features of the platforms used to make the application. The data stored on the flat files [10] can only be accessed by the IP (Internet Protocol) addresses that have been whitelisted by the administrator.

The following is the first form which is shown to the user and it marks the beginning of the procedure. It is present at the emigration official's desk and he/she is required to fill in the details. The application ends with a form at the immigration official's desk who rectifies all the information and mark the journey as complete. He/she also marks that an official immigration has taken place by undertaking the right procedures.

Fig. 3. The form used to store the information of a person at the state of immigration as well as emigration.

The form shown in the above image is required to be submitted during emigration. The submission of the form initiates a transaction which is validated using Proof of Work. In the case study to illustrate the proposed algorithm, Rinkby test network has been employed and Metamask is used to pay gas price required during transaction validation. As explained before, the application has been employed in the test network so no real money is denominated from the individual's bank account.

Fig. 4. Form to get the information from the Blockchain.

The state of immigration process as well as personal details of an individual can be checked using his/her identification information. In this case a transaction is not made as this function requires the reading of the record in blockchain. The information returned can be matched with the official records of a suspect.

When a transaction is not made, it means that no money is deducted from the user's account. This is because no change of values is performed at any place. Also, there is no need for performing any calculations, which would have needed compute memory and Arithmetic and Logical instructions to be undergone, requiring money to be transferred and transaction to be made.

In this case, the details of the individual are read from their corresponding records and the official is required to check the person's details and confirm its validity hence, there is no need for performing any transactions. The data from the Blockchain is read and relayed to the official as-it-is.

The next step is to update the migrant's status to have safely reached the intended destination. This would be done by the immigration's official present at the booth.

Fig. 5. Form to mark the successful arrival of the migrant at the intended destination.

As mentioned above, an individual's information needs to be updated at the time he/she enters another country. Due to security reasons, only the information about successfully reaching the destination can be changed. This is because more power given to the application will create security issues. Moreover, limited power means that there will not be a single point of failure. Even if the application fails, no one will be able to update the information of an individual due to the limited rights of the software. Therefore, at its worse, the application would not accept any data to be entered rendering it safeguarded against miscreants.

Fig. 6. AWS S3 server with the stored data.

The above figure explains how the data is stored in the AWS S3 storage solution looks. This figure gives image of the page that shows the overview of all the files stored at the storage solution. Currently, there is only one file in the bucket which has been stored for the purposes of demonstrating the application. As mentioned before, this service stores the flat files of the user i.e. the legal and authorized documents obtained from the government that define the person's identity.

The usage of Blockchain technology for this application and the research, ensures the safety of data inserted into the system. Owing to its decentralized storage ability, Blockchain forms copies of the data and stores it at various places. This eliminates the fear of mutating the data, especially when it stores the personal information of an individual. Also, only the user with the administrator privileges will be able to manipulate the data of the system and store details and other important files of the migrants.

The following section discusses the various technologies and services that could be implemented in the system once it has been deployed in the real world scenario. Here, we discuss about all the technologies that would be added into the system when it would be working at various ports, in a decentralized manner.

V. POTENTIAL IMPLEMENTATION

This section discusses a very important aspect of the application: The implementation of the software in the real world. This means that the section discusses about various potential technologies that could be integrated to the application once it has been deployed to the real-world framework. The mentioned technologies would be implemented when the application is working at various ports of immigration and emigration at which it is supposed to be implemented upon.

Two technologies have been discussed in this section to reiterate over the potential importance of the decentralized operability of the application in the real world:
1. AWS S3 Versioning [11]
2. AWS Patch Manager [12]

A. S3 Versioning

With the feature of Versioning enabled for an AWS S3 bucket, it allows to keep multiple versions of an object in the same bucket. This feature will prove to be helpful in case of intentional or unintentional mutation of the documents stored in S3. With versioning, we can easily recover from both unintended user actions and application failures.

In one bucket, for example, we can have two objects with the same name, but different version IDs, such as photo.gif (version 111111) and photo.gif (version 121212).

Fig. 7. AWS S3 Versioning enabled.

Versioning may be useful in various scenarios. A few examples are listed:

- In case if the document gets deleted, instead of removing it permanently, AWS S3 inserts a delete marker, which essentially means that the deleted object can be recovered.
- If the document gets overwritten, it results in a new version of the document in the bucket. The previous version of the object can always be restored from both console as well as CLI.

The versioning state will be applied to all of the documents in the S3 bucket. The first-time versioning is enabled for a bucket, documents stored in it thereafter will always be versioned and given a unique version ID.

As a general case,
- Objects stored in the bucket before setting the versioning state have a version ID of null.
- When versioning is enabled for the first time, existing objects in a bucket do not change.
- Versioning cannot be stopped once it is enabled, it can only be suspended.

Along with the versioning, MFA (Multifactor Authentication) Delete can also be applied to the objects in the bucket as doing it will ensure that the documents are not deleted unintentionally or illegally.

Moreover, enabling the versioning feature in AWS S3 is one of the requirements to make the data stored in the bucket highly available by the means of bucket replication.

The following image represents the console of the administrator after the technique of S3 Versioning has been applied. This represents the overview application.
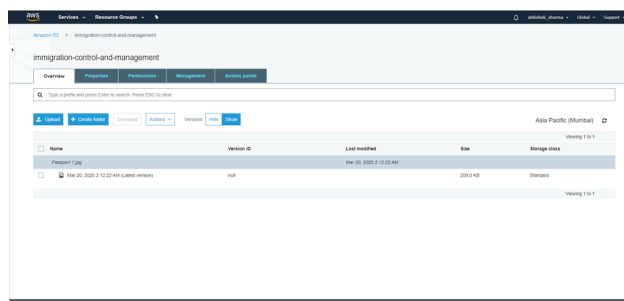

Fig. 8. Console overview after enabling AWS S3 Versioning.

B. AWS Patch Manager

The service, Patch Manager, provided by the Amazon Web Services, can help to maintain the consistency for both operating systems and applications. It will ensure proper functioning of the proposed application. It would help to patch all the servers responsible for running the application concurrently with a click of a button. This will not only keep all the servers in the same state but will also ensure that there is no discrepancy in the way the data is processed.

Moreover, Patch Manager integrates with AWS Identity and Access Management (IAM), AWS CloudTrail, and Amazon CloudWatch Events to enhance its security and experience that includes real-time event notifications and the ability to audit usage of application.

Patch Manager uses patch baselines, which include rules for approving the worthy patches as well as a list of approved and rejected patches. The patches can be applied on a regular basis by scheduling patching to run as a Systems Manager maintenance window task.
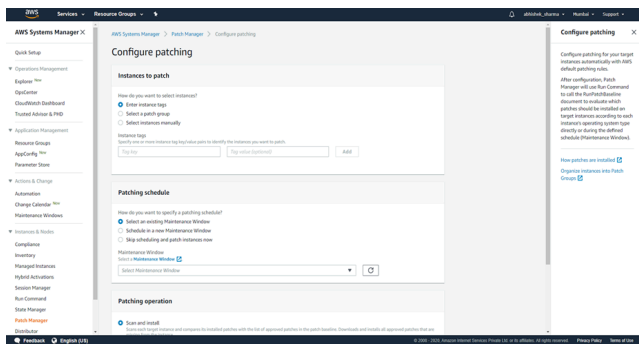

Fig. 9. Console overview after enabling AWS Patch Manager.

The above image gives an overview of how the console of the user of this application (the administrator) would look like when the technique of AWS Patch Manager has been applied. As mentioned before, this is only possible if the software has been deployed at various ports and the ports are needed to be synchronized.

The following section describes the various conclusions that have been derived by implementing the given system on a relatively small scale i.e. on a test network. As mentioned before, the system has been deployed and the transactions have been executed to imitate the real-world scenario but no real money was used to demonstrate the implementation of the software.

VI. CONCLUSIONS

As seen in the previous sections, the proposed system can store the details of all the people who depart their country at any time. This is done during the process of emigration. By matching the records of the Blockchain with the official records of the individual, we can find all the illegal immigrants in the country. The system can tell if a migrant has gone missing by mentioning the records who left their native country but did not deport their destination. This can be achieved by verifying the statistics of immigration and emigration.

Also, we can identify the travelers who entered a country but did not leave their native land as their details will not be found in the system at the time of their immigration. This

ensures that every individual, leaving the country has a valid identity and is entering the new nation with the valid identity.

Finally, as promised by the application, all flat files are being stored in the AWS S3 storage service that is a proprietary service provided by Amazon Web Services (AWS). Flat files of an individual include the various legal documents and official material that is provided to the individual by their respective government. The benefit of storing these files in such a way is that these files are not accessible to the general public. Only whitelisted IP (Internet Protocol) addresses are allowed to access these files or upload them. The durability of the application increases manifolds as the service is proprietary. Moreover using a licensed service like Amazon Web Services' storage servers leads to an enhanced security and reliability of the application as opposed to using an open source platform, which can be compromised leading the information into malicious hands.

This project was made possible by exploiting the merits of Blockchain technology. The existence of Blockchain on Hyperledger [13] makes it possible for all the instances of this application to be on a unified decentralized platform. Here, the government agencies of security would be safeguarding the ports to their country by working together in harmony.

VII. SCOPE OF IMPROVEMENT

While the proposed system utilizes the immutability and decentralized storage ability of Blockchain technology, it still has some limitations that could be overcome.

1. The proposed system has been made by inserting only primitive information about the traveler. This has left room for improvement. There can be an addition of some migrant specific data like a biometric system that would take the data of every migrant's biometric [14] and would store it along with the personal information.
2. This system also lacks other parameters that can contribute to its safety and robustness. The current parameters include only the native and destination land of the individual. Improved database may contain information about their past and if they have been involved in any illegal resettlement before, which would aid the admissions officer.
3. The administrator handles all the entries of data. So, if the administrator is compromised [15], the system may get into wrong hands. This way fictitious data could be entered into the system which would be impossible to discern from the legitimate information.
4. Load balancer [16], [17] can be deployed in front of the servers pertaining to this application. It would allow for the proper distribution of the traffic that would be generated by the large number of machines, running on almost all of the ports, the application to ensure that none of the servers are overloaded with requests which can lead to unpredictable delays.
5. Moreover, adding the load balancer will also act as a firewall for the system's server and would do the job of Intrusion Detection System [18]. Therefore, it would be able to monitor network traffic for suspicious activity

and issues alerts when such activity is discovered.

APPENDIX: TECHNOLOGIES, SOFTWARE AND HARDWARE

This section describes the additional information necessary for implementing the application to conduct research and accomplish the project. This paper required an application to be implemented, and that application was tested and researched upon to find the results obtained here. This information is the technologies, software and hardware used to develop the necessary program.

A. *Technology used*
- Blockchain - Blockchain is a type of distributed ledger for maintaining a permanent and tamper-proof record of transactional data.
- Ethereum - Ethereum is a global, open-source platform for decentralized applications.
- React – React is a JavaScript library that is used to build interactive front end web applications.
- Node - Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications and its applications are written in JavaScript

B. *Hardware Used:*

Computers with the following specifications was used for the construction of the application:
- Processor: Ryzen 5 1600X
- GPU: AMD RX580
- RAM: 8GB
- Internet connection

C. *Software Used:*
- Atom Code Editor
- Chrome V8 Engine
- Metamask for local testing
- Various libraries like Ganache, Web3, and Express

REFERENCES

[1] International Organization of Migration. Section 3.1 Passport and VISA system. [Online] Available:http://www.rcmvs.org/documentos/IOM_EMM/v3/V3S01_CM.pdf

[2] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System".

[3] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum project yellow paper, vol. 151, pp. 1–32, 2014.

[4] Blockchain Revolution – 2016, By Don Tapscott and Alex Tapscott.

[5] Transfer e-Borders data: general aviation and maritime. https://www.gov.uk/government/publications/transfer-e-borders-data-general-aviation-and-maritime. Accessed 20 Feb 2018

[6] Wurster, S., et al.: Specification on blockchain technology. ISO/TC 307, Tokyo (2017).

[7] How blockchain can create a more humane process for refugees (article) July 13, 2018 https://bigthink.com/reuben-jackson/how-blockchain-will-help-refugees-and-migrants

[8] How Blockchain can benefit migration programmes and migrants (article) February 22, 2018 https://migrationdataportal.org/blog/how-blockchain-can-benefit-migration-programmesand-migrants

[9] Smart Contracts Explained [Online] Available: http://www.blockchaintechnologies.com/blockchain-smartcontracts

[10] Sunil Gulabani, Hands-on Simple Storage Service (S3), Practical Amazon EC2, SQS, Kinesis, and S3 pp 223-307, 2017 – Springer.

[11] AWS S3 versioning documentation explaining the overview [Online] Available: https://docs.aws.amazon.com/AmazonS3/latest/dev/Versioning.html

[12] AWS Patch Manager documentation explaining the overview [Online] Available:https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-patch.html

[13] Hyperledger Fabric White paper. [Online] Available: http://www.theblockchain.com/docs/Hyperledger%20Whitepaper.pdf

[14] Ruggero Donida Labati, Angelo Genovese, Enrique Muñoz, Vincenzo Piuri, Fabio Scotti, Gianluca Sforza, "Biometric Recognition in Automated Border Control", ACM Computing Surveys, vol. 49, pp. 1, 2016, ISSN 03600300, Available: http://piurilabs.di.unimi.it/Papers/csur_2016.pdf

[15] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang, An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends, 2017 IEEE 6th International Congress on Big Data.

[16] Deepak Vohra, Load Balancing, Docker Management Design Patterns, 2017 – Springer, pp 219-239.

[17] Hasan, Raed A; Mohammed, Muamer N; Ameedeen, Mohammed Ariff Bin; Khalaf, Emad Taha, Dynamic Load Balancing Model Based on Server Status (DLBS) for Green Computing, Advanced Science Letters, Volume 24, Number 10, October 2018, pp. 7777-7782

[18] Taha Arian, Amir Kusedghi, Bijan Raahemi, Ahmad Akbari, A Collaborative Load Balancer for Network Intrusion Detection in Cloud Environments, Journal of Computers Volume 12, Number 1, January 2017.