# CRAB: Blockchain Based Criminal Record Management System

**Conference Paper** · December 2018

DOI: 10.1007/978-3-030-05345-1_25

# CRAB: Blockchain Based Criminal Record Management System

Maisha Afrida Tasnim[1], Abdullah Al Omar[1](✉),
Mohammad Shahriar Rahman[2], and Md. Zakirul Alam Bhuiyan[3]

[1] Department of Computer Science and Engineering, University of Asia Pacific,
Dhaka 1215, Bangladesh
maishaafrida@hotmail.com, omar.cs@uap-bd.edu

[2] Department of Computer Science and Engineering,
University of Liberal Arts Bangladesh, Dhaka 1209, Bangladesh
shahriar.rahman@ulab.edu.bd

[3] Department of Computer and Information Sciences, Fordham University,
New York, NY 10458, USA
mbhuiyan3@fordham.edu

**Abstract.** Criminal records are highly sensitive public records. By incorporating criminal records in a blockchain, authenticity and rigidity of records can be maintained; which also helps to keep the data safe from adversaries. A peer to peer cloud network enables the decentralization of data. It helps prevent unlawful changes in the data. This paper introduces a criminal record storage system by implementing blockchain technology to store the data, which helps to attain integrity and security. Our system presents ways in which the authority can maintain the records of criminals efficiently. Authorities (e.g., Law enforcement agencies and courts) will be able to add and access criminal data. General users (e.g., selected organizations and/or individuals, airports, visa application centers etc.) will have access to the data so that they can look up criminal records. Proper and timely access to authentic criminal records is essential to enforce the law. The effect of corruption on the law enforcement forces will also decrease, as this will cut off an entire scope of corruption by removing any possibility of tampering with criminal records data by thorough accountability.

**Keywords:** Criminal records · Blockchain · Authenticity
Cloud network · Decentralization · Law enforcement

## 1 Introduction

A chief function of the government is to preserve data about individuals. Administering and utilizing these data can prove to be cumbersome, even for advanced governments. Different government law enforcement agencies have separate databases, which creates a barrier in the fluidity of data flow between different government agencies. The existence of such multiple databases also

G. Wang et al. (Eds.): SpaCCS 2018, LNCS 11342, pp. 294–303, 2018.
https://doi.org/10.1007/978-3-030-05345-1_25

increases the cost of their security and thus, the probability of unlawful changes are increasing gradually [1].

With the growing size of records, a good record keeping and information sharing system has become necessary in todays global environment. Law enforcement agencies have to communicate between themselves and across countries in order to keep national security intact. Having accurate and time stamped records makes it easier to accomplish the mission [2].

This is where blockchain comes into the picture. The blockchain ledger ensures no single party can control the peer to peer network so the risk of data tampering is abating. In addition, the dispersed characteristic of the blockchain ledger means that it is extremely difficult to break and also the risk of information being meddled with is greatly reduced compared to current systems that use traditional digital databases [3]. One of the aims of our system is to ensure that evidence information is not tampered during court proceedings by storing the data in cloud and keeping the transaction log and provenance data in blockchain.

A central database can be subjected to many types of hacks, most of which may severely damage the integrity and validity of the data. The security of the system depends on the database system itself. SQL injection attacks have become more common in recent days [4]. SQL injection is a highly destructive attack in which hackers try to access information stored in a database. The decentralized nature of blockchain guarantees that inherent problems of the system, like hardware and software malfunctions, have no effect on integrity of the data, as the data has multiple copies stored on each node of the network. Data in blockchain is immutable, implying that any and all changes are clearly visible on the entire network. Data updated by a node is verified by multiple nodes, and thus falsified data can seldom find its way into the blockchain [5]. Any attempt to destabilize the system will have to include simultaneous attacks on at least 51% of nodes of a certain blockchain to affect a single block. This decreases the chance of attacks exponentially with the increasing number of nodes [6].

Our system uses a decentralized data management process. The users of the system are pre-registered. Data senders must sign in to the system first. Then they digitally sign the data. The digital signature is verified by the system to make sure the data is authentic. The verified data is encrypted with a randomly generated encryption key and is sent to the cloud data storage. The metadata of this transaction is sent to the blockchain. The location of this data on the blockchain is retrieved by the system. The system then stores essential searching parameters, like case number, name of offender, passport number and national identification number in a local database. The encryption key and location of the data on the blockchain is also stored on the local database. Data receivers also have to login to the system. Then they can search for data using the aforementioned parameters. The system fetches the data and decrypts it. The system then adds this data retrieval event to the blockchain as a transaction and forwards the decrypted data to the data receiver. Even if any adversary gains access to the encryption key, they can possibly just view the data. They cannot modify data since data upload requires a valid digital signature from a pre-registered user. Also, any change to the data will be recorded on the blockchain as a transaction.

**Our Contribution.** Our proposed system stores an individual's criminal records. The purpose of our system is to ensure that the stored information is secure and cannot be accessed or altered by attackers. Currently large amounts of data is stored in databases which makes it highly vulnerable to attacks. Databases might as well crash, resulting in loss of data. In our system such problems will not arise since we use blockchain to store the data transaction logs alongside encrypting the data so it cannot be altered. Each node will have a copy of the transaction logs [7]. The data itself will be stored in a decentralized cloud system. Decentralization increases redundancy of the data. Elliptic Curve Cryptography (ECC) [8] encryption scheme is used in our platform to encrypt the criminal data. We generate the digital signature according to Schnorr digital signature scheme [9].

Our system uses a data provenance architecture. Information regarding upload, access or changes in Cloud data is stored in the blockchain which ensures security, privacy and integrity [10]. These security parameters are crucial while dealing with such sensitive data. CRAB makes the stored information accessible to courts, selective government organizations and individuals, all police stations, visa application centers, airports etc.

**Organization of the Paper:** The remainder of the paper is organized as follows: Sect. 3 outlines the protocol and details the steps, Sect. 4 briefly analyses the features of the protocol, and Sect. 5 includes some concluding statements on the probable outcome of the implementation of such a system.

## 2 Related Work

Various data sharing systems using blockchain have been developed [11]. Research work has been done on cloud data provenance architecture. Two such platforms are ProvChain [10] and SmartProvenance [12]. ProvChain is a decentralized cloud data provenance architecture that uses blockchain technology. When a user accesses data from the cloud, records are kept in the blockchain as transactions. It ensures that the records cannot be tampered. In ProvChain, the provenance auditor endorses provenance data by fetching transactions from the blockchain network by using blockchain-receipt which contains data in block and transactional information [10]. Here the Provenance Auditor (PA) cannot be fully trusted. Since PA has access to both user and provenance data; it can cause devastating damage to the system. To avoid this, the data is encrypted before uploading to the cloud. As such, the PA cannot directly access the data without the decryption key [10]. The SmartProvenance system is built on the existing Ethereum system, which uses smart contracts. These are used to store metadata of a file and include an event log. The event log is an immutable record consisting of the changes made to the file or data. This system can only guarantee honest behavior if at least half of the users able to access the data and provenance are honest. There also must exist a secure platform for exchanging external keys among the users, so a user can provide access to other users [12].

Work based on the Etheruem system is efficient, as mentioned in Forensic-Chain [13], where the smart contracts help to do most of the transaction verification work [13]. Similar to SmartProvenance and Forensic-Chain, MedRec [14] is a system which is also based on Ethereum. However, it uses Ethereum smart contracts to allocate each block to a single file's access permission data and state transition records. This system primarily focuses on securing access permissions to medical records through the blockchain [14].

The UK Ministry of Defence (MoD) agency is considering the use of blockchain to improve the reliability of a network that uses sensors to track national concerns. HoustonKemp, a Singapore based tech firm is working in Australia to develop a blockchain based, reliable and feasible system that will be used to keep records of all investigative intelligence [3].

Controllability and traceability are key features of a privacy preserving system [15]. Our system rules out any human intervention from data storage, integrity, privacy and traceability aspects. Ethereum based smart contracts facilitate our systems functionality, making the process of designing and implementing the blockchain simple. This also allocates the mining tasks to an existing market, generating less cost. Blockchain in government [7] talks about how the government can benefit from various applications of blockchain technology.

## 3 CRAB-Protocol

In this section we demonstrate the design and architecture of our system. Table 1 shows the notations that are used in this section.

**Table 1.** Terminology table

| Notation | Description |
|---|---|
| ID | Data sender's ID |
| PWD | Data sender's password |
| $\mathrm{U_D}$ | Criminal data uploaded by sender |
| $\mathrm{V_D}$ | Verified criminal data |
| $\mathrm{T_D}$ | Transaction data |
| $\mathrm{B_{id}}$ | Block number where meta data of transaction is saved |
| CID | Criminal identification data |
| UAD' | Consists of CID, $\mathrm{B_{id}}$, and Enc(Key) |
| $\mathrm{ID_X}$ | Sender X's ID |
| $\mathrm{PWD_X}$ | Sender X's password |
| $\mathrm{U_{DX}}$ | Criminal data uploaded by sender X |
| $\mathrm{V_{DX}}$ | Verified data of sender X |
| $\mathrm{T_{DX}}$ | Transaction data of sender X |
| $\mathrm{B_{idX}}$ | Block number where transaction data of user X is saved |

### 3.1 Protocol Entities

Figure 1 shows the high-level view of our system. The entities and their roles in the system is described below.

**Data sender** is the authorized personnel from Police station, court, law enforcement agencies and armed forces, who will have to store criminal record and information into the system. The data will be verified using the sender's digital signature, and then encrypted and stored in Data Storage along with CID.

**Data Receiver.** Data senders, airports, visa application centers and selected organizations will play the role of data receiver in our system, they will have to sign in to the system and request for accessing data from the system using CID.

**Functional Unit (FU)** is the most important part of our system. This module authenticates users, verifies and encrypts data. It sends the data to the Data Storage after encrypting it. This unit also retrieves the $\mathrm{B_{id}}$ corresponding to a transaction. When a user has to retrieve data from the cloud, it accesses the FU and requests for data using CID. FU interacts with Local server, blockchain and Data Storage to fetch the data and it sends decrypted data to the user.

**Data Storage** is a part of the global cloud. It receives encrypted data from FU. Data storage stores encrypted criminal data and CID.

**Blockchain** stores the meta data of transactions($\mathrm{T_D}$). It is present on the cloud. It can only be accessed directly by the FU. We are using Ethereum based permissioned blockchain, which ensures data security since only FU will have the permission to access the blockchain.

**Local Server** stores UAD' which comprises of CID, $\mathrm{B_{id}}$, and Enc(key). It sends UAD' to FU upon request.

### 3.2 Steps Involved

- **Step-1.** The sender accesses the system using their ID and PWD.
- **Step-2.** The sender digitally signs the $\mathrm{U_D}$ and sends it to the FU.
- **Step-3.** $\mathrm{V_D}$ is encrypted and sent to **Data Storage** along with CID.
- **Step-4.** $\mathrm{T_D}$ is sent to **blockchain**.
- **Step-5.** $\mathrm{B_{id}}$ is sent to the FU from **blockchain**.
- **Step-6.** UAD' is sent to the **Local Server**.
- **Step-7.** The receiver accesses the system with ID and PWD
- **Step-8.** FU requests Local Server for data with CID.
- **Step-9.** UAD' is returned to FU.
- **Step-10.** FU requests for data from **Data Storage** with CID.
- **Step-11. Data Storage** sends encrypted data to FU.
- **Step-12.** $\mathrm{T_D}$ is sent to **blockchain**.
- **Step-13.** $\mathrm{B_{id}}$ is returned to FU.
- **Step-14.** FU sends updated UAD' to **Local Server**.
- **Step-15.** Decrypted data is sent to the receiver.

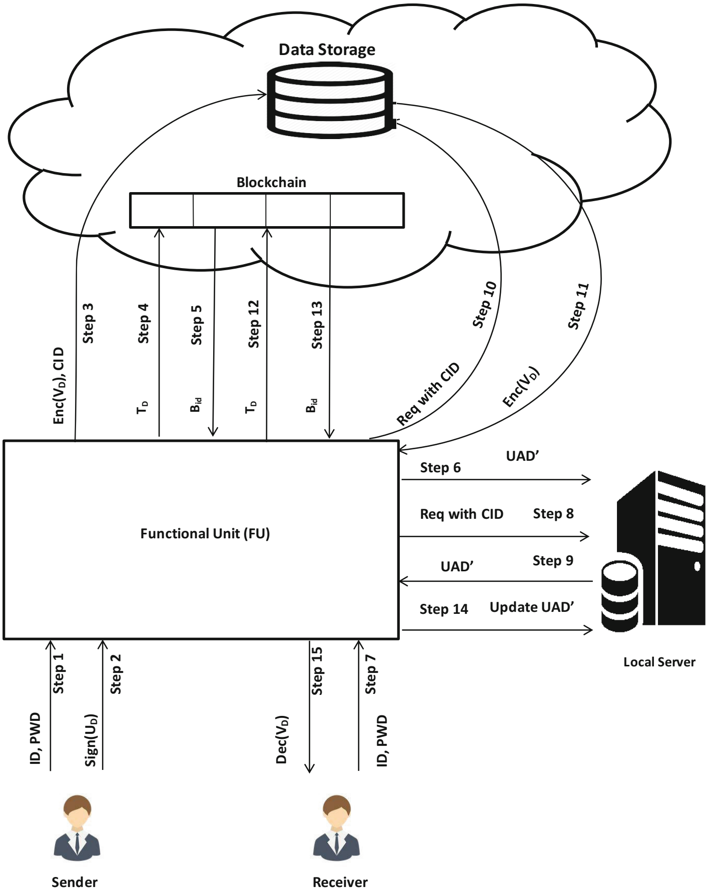

**Fig. 1.** Overview of the CRAB protocol

### 3.3 Formal Description of Protocol

In this section we describe how Data Sender, Data Receiver, and our system interact with each other while sending and receiving data. For any transaction in our system, parties need to be pre-registered. Any data transmission from parties who are not registered will be ignored by the system.

**Protocol Between Data Sender and System**
Police station, court, law enforcement agencies and armed forces will play the role of data sender in this protocol. Digitally signed data will be verified; the data will then be encrypted and sent to the Data Storage. The equation for key generation can be written as:

$$KeyGen(Random) = Enc(Key) \tag{1}$$

Suppose a Data Sender X wants to upload a file to the Data Storage. X signs in to the system with $\mathrm{ID_X}$, and $\mathrm{PWD_X}$. X digitally sign the data ($\mathrm{U_{DX}}$) and send it to FU. FU verifies the digital signature against the data. A random encryption key is generated, $\mathrm{V_{DX}}$ is encrypted according to the following equation:

$$Enc(V_{\mathrm{DX}}, Key) = Enc(V_{\mathrm{DX}}) \tag{2}$$

$\mathrm{V_{DX}}$ and UAD is sent to Data storage. The meta data of X's transaction, $\mathrm{T_{DX}}$ is sent to the blockchain. The location of the transaction data on the blockchain, $\mathrm{B_{idX}}$ is sent to FU. UAD' is sent to Local Server to be used for accessing data. Upon receiving $\mathrm{B_{idX}}$ the sender can be assured that the data has been uploaded successfully to the Data Storage.

**Protocol Between System and Data Receiver**
Suppose X is a Data Receiver. X uses $\mathrm{ID_X}$ $\mathrm{PWD_X}$ to sign in to the system, FU requests for data using CID, the Local Server returns UAD' to FU. FU requests for data from Data Storage. Data Storage returns encrypted $\mathrm{V_D}$ to FU. FU decrypts $\mathrm{U_D}$ according to the following equation:

$$Dec(Enc(V_{\mathrm{D}}), Key) = V_{\mathrm{D}} \tag{3}$$

The decrypted data is then sent to the receiver.

## 4 Protocol Analysis

- **Integrity:**
  - **Authentication data integrity:** Only pre-registered users will be able to enter or retrieve data. Data sender X and receiver Y first need to authenticate themselves. They will have to use ID and PWD provided by the authority, which are stored in the **Local server**. When X or Y provides ID and PWD, the system retrieves the actual ID and PWD from the **Local server**; if the user provided ID and PWD matches with the retrieved ID and PWD, the user is granted access to the system. Therefore authentication data is only know to X, Y and the system.
  - **User data integrity:** Using the encryption function below the criminal data is encrypted.

$$Enc(V_{\mathrm{DX}}, Key) = Enc(V_{\mathrm{DX}}) \tag{4}$$

This ensures data integrity since the data stored in the **Data Storage** will not make any sense to anyone except for the data sender X. If X or Y requests for the data, the **FU** retrieves the data from the **Data Storage** and decrypts it using the following equation:-

$$Dec(Enc(V_{\mathrm{D}}), Key) = V_{\mathrm{D}} \tag{5}$$

To break this integrity level adversaries need to break the ECC encryption scheme.

- **Accountability:** Transactions in blockchain help to monitor data changes. When the data sender X with $\mathrm{ID_X}$ sends $\mathrm{V_D}$, the metadata $\mathrm{T_{DX}}$ is recorded in the blockchain.

$$Transact(ID_{\mathrm{X}}, V_{\mathrm{D}}, Current\_time, Set) = T_{\mathrm{DX}} \tag{6}$$

  When a data receiver Y with $\mathrm{ID_Y}$ attempts to retrieve $\mathrm{V_D}$, this access $\mathrm{T_{DY}}$ is also recorded as in the blockchain.

$$Transact(ID_{\mathrm{Y}}, V_{\mathrm{D}}, Current\_time, Get) = T_{\mathrm{D}} \tag{7}$$

  Due to the blockchain transactions, all data senders and receivers are accountable for any interaction with the data on the cloud.
- **Security:** Data is stored in an encrypted form and cannot be accessed without the encryption key which is stored separately.
  When X sends $\mathrm{U_D}$, the **Functional Unit(FU)** verifies and encrypts it. $\mathrm{V_{DX}}$ and $\mathrm{Enc(V_{DX})}$ are not visible to X and is completely handled by the **FU**.
  When Y requests to retrieve $\mathrm{V_{DY}}$, the **FU** decrypts $\mathrm{Enc(V_{DY})}$ and forwards it to Y. Y is privy to any access in the system except its initial request. So the data is completely secure and void of direct access by sending receiving entities.
- **Automation:** The system is totally automated and requires no human intervention, which reduces risk of error.
- **Sustainability:** Since the system is automated; there is a very low risk of errors occurring. Our platform uses tried and tested methods of encryption. Thus, the system is sustainable.

**Table 2.** Summary table

| Feature | Blockchain | Traditional database |
|---|---|---|
| Storage | Decentralized | Centralized |
| Mutability | Immutable | Mutable |
| Redundancy | Redundant | Non-redundant |
| Cost | Decreasing cost for increasing amount of data | Cost increases with increasing data size |
| Transparency | All nodes attest to validity of data | Validity of data may only be checked by database administrator |
| Point of failure | Fails only if all nodes simultaneously fail | Failure may result from any hardware or software failure of the server machine |
| Interoperability | Good interoperability | Hard to achieve interoperability |

Table 2 shows the comparisons of features using blockchain and traditional databases. Storage refers to the manner in which the data is stored. Mutability is the ability of data to be changed. Redundancy refers to whether the data can be easily recovered if lost. Cost refers to the financial cost of implementing and maintaining these systems. Transparency means whether the data activity in the systems is visible or not. Point of failure indicates the weakest attribute of the system that can be used to destabilize or destroy it. Interoperability refers to communication between multiple similar systems.

## 5 Conclusion

Public records often are tampered with, and their effects are adverse. Our system lets us remove all such problems by means of decentralized data storage. Digital signatures confirm the authenticity of uploaded data. Each data sender bears the complete responsibility of the data contents. Encryption furthers the security objective of this system. The randomly generated encryption keys ensure that no two files have the same key, which exponentially reduces the risk of attacks. The cloud components, which are data storage and blockchain, are not directly accessible by any user. All these together ensure maximum security of data and precise provenance recording, and also helps overcome other possible software/hardware failure issues. Further research on this topic can bring a whole scale implementation in a city, region, state or even country.

## References

1. Cheng, S., Duab, M., Domeyer, A., Lnudqvis, M.: Using blockchain to improve data management in the public sector. https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/using-blockchain-to-improve-data-management-in-the-public-sector
2. Ariq, M., Shakeel, S., Ali, Z.: Report on criminal record management system. https://www.slideshare.net/hashimabbasi786/criminal-recordmanagementsystem-report
3. Open Trading Network: UK police - blockchain solutions on the horizon. https://medium.com/@otncoin/uk-police-blockchain-solutions-on-the-horizon-60e3e1932ef3
4. Thoms, N.: SQL injection: still around, still a threat. https://www.fasthosts.co.uk/blog/digital/sql-injection-still-around-still-threat
5. Anh, D.T.T., Zhang, M., Ooi, B.C., Chen, G.: Untangling blockchain: a data processing view of blockchain systems. IEEE Trans. Knowl. Data Eng. **30**(7), 1366–1385 (2018)
6. Miles, C.: Blockchain security: what keeps your transaction data safe? https://www.ibm.com/blogs/blockchain/2017/12/blockchain-security-what-keeps-your-transaction-data-safe/
7. Ølnes, S., Ubacht, J., Janssen, M.: Blockchain in government: benefits and implications of distributed ledger technology for information sharing (2017)
8. Setiadi, I., Kistijantoro, A.I., Miyaji, A.: Elliptic curve cryptography: algorithms and implementation analysis over coordinate systems. In: 2015 2nd International Conference on Advanced Informatics: Concepts, Theory and Applications (ICAICTA), pp. 1–6. IEEE (2015)

9. Boneh, D.: Schnorr digital signature scheme. In: van Tilborg, H.C.A., Jajodia, S. (eds.) Encyclopedia of Cryptography and Security, pp. 1082–1083. Springer, Boston (2011). https://doi.org/10.1007/978-1-4419-5906-5
10. Liang, X., Shetty, S., Tosh, D., Kamhoua, C., Kwiat, K., Njilla, L.: Provchain: a blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. In: Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp. 468–477. IEEE Press (2017)
11. Crosby, M., Pattanayak, P., Verma, S., Kalyanaraman, V.: Blockchain technology: beyond bitcoin. Appl. Innov. **2**, 6–10 (2016)
12. Ramachandran, A., Kantarcioglu, M.: Smartprovenance: a distributed, blockchain based dataprovenance system. In: Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy, pp. 35–42. ACM (2018)
13. Lone, A.H., Mir, R.N.: Forensic-chain: ethereum blockchain based digital forensics chain of custody. Sci. Pract. Cyber Secur. J. (2018). ISSN 2587-4667. https://journal.scsa.ge/issues/2017/12/783
14. Azaria, A., Ekblaw, A., Vieira, T., Lippman, A.: MedRec: using blockchain for medical data access and permission management. In: International Conference on Open and Big Data (OBD), pp. 25–30. IEEE (2016)
15. Al Omar, A., Rahman, M.S., Basu, A., Kiyomoto, S.: MediBchain: a blockchain based privacy preserving platform for healthcare data. In: Wang, G., Atiquzzaman, M., Yan, Z., Choo, K.-K.R. (eds.) SpaCCS 2017. LNCS, vol. 10658, pp. 534–543. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-72395-2_49