

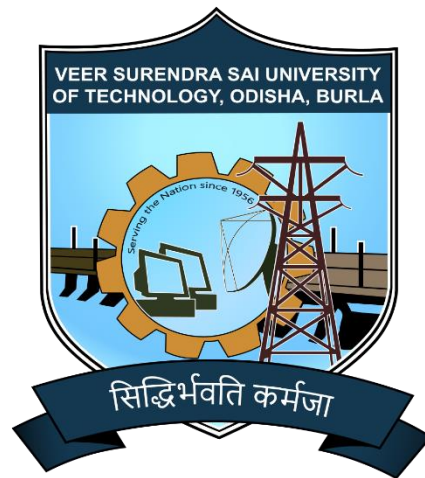
Blockchain Based Chain Of Custody

Sourav Mahapatra

1902041036

Sushruta Behera

1902041051



COMPUTER SCIENCE AND ENGINEERING

VEER SURENDRA SAI UNIVERSITY OF TECHNOLOGY, BURLA

2022-23

Blockchain Based Chain Of Custody

Sourav Mahapatra

Regd.No.-1902041036

Sushruta Behera

Regd.No.-1902041051

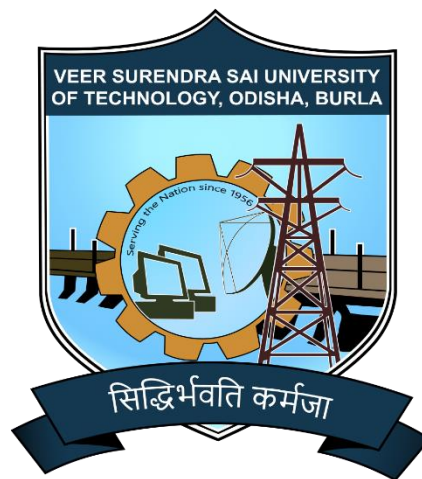
Supervisor

Dr.Sumitra Kisan

Assistant Professor,

Dept. of Computer Science Engineering

VSSUT, Burla



COMPUTER SCIENCE AND ENGINEERING

VEER SURENDRA SAI UNIVERSITY OF TECHNOLOGY, BURLA

2022-23

Department of Computer Science and Engineering
VEER SURENDRA SAI UNIVERSITY OF TECHNOLOGY
Burla, Odisha



Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Date:06/05/2023

Sourav Mahapatra
Regd.No.-1902041036

Sushruta Behera
Regd.No.-1902041051

Department of Computer Science and Engineering
VEER SURENDRA SAI UNIVERSITY OF TECHNOLOGY
Burla, Odisha



Certificate

This is to certify that the project entitled “Blockchain Based Chain Of Custody” submitted by Sourav Mahapatra and Sushruta Behera bearing Registration no. 1902041036 and 1902041051 of Department of Computer Science & Engineering, VSSUT, Burla, Odisha, India for the award of the degree of Under Graduation in Computer Science and Engineering, is a record of an original research work carried out by him under my supervision and guidance.

Prof.(Dr)SuvasiniPanigrahi
Head of Department

Dr. Sumitra Kisan
(Supervisor)

Acknowledgement

We would like to express our sincere gratitude to our supervisor, Dr. Sumitra Kisan, for her invaluable help during the course work towards this dissertation. She was a source of constant ideas and encouragement and provided a friendly atmosphere to work in. We are really very thankful to her for everything.

We are also thankful to Dr. Suvasini Panigrahi, Head of the Department and to all the faculties of Department of Computer Science and Engineering for having supported us to carry out this dissertation and for their constant advice. We would like to thank all our friends for their encouragement and understanding. We would like to express our heart-felt gratitude to them.

Approval Sheet

This dissertation entitled, “Blockchain Based Chain Of Custody” by Sourav Mahapatra and Sushruta Behera are approved for the degree of Bachelor of Technology in Computer Science Engineering, Department of Computer Science and Engineering.

Examiner

Supervisor

Date: 06/05/2023

Place: Burla

ABSTRACT

The digital evidences consist of important information about the criminal activity such as EvidenceID ,description about the evidence on the basis of which important decision are taken at the court. Before the evidence will reach to court it passed through a number of persons such as(police, forensic doctor,lawyer)etc so there is a main issue that it can be tempered or modified on it's way .So there is a process known as Chain Of Custody(CoC) which ensure that evidences would not be altered during the investigation process.CoC is managed manually with entries .In this paper we propose a Blockchain-based Chain Of Custody (B-CoC) in which we store the evidenceID,description about the evidence on blocks .B-CoC guarantees integrity of evidence(It means the evidence can't get modified) ,traceability(It provide us complete information about the owners who were accessing blocks and in what order).

Keywords:

Digital Forensics, Chain of custody, digital evidence, Ethereum Virtual Machine(EVM).

CONTENTS

1.Introduction	1
1.1. Overview	1
1.2. Objectives	3
1.3.Problem Statements	3
2.Literature Survey	4
3.Existing Work	7
4.Proposed Work	10
4.1. Overview	10
4.2. Algorithm	12
4.3. Flowchart	13
5.Result and Discussion	19
6.Conclusion and Future Scope	27
7.References	28

LIST OF FIGURES

1.Blockchain	10
2. B-CoC Architecture	9
3.Evidence Creation	13
4.Evidence Transfer	14
5.Evidence Display	15
6.Evidence Remove	16
7.Hash Generator	17
8.Detection of Error Due to change in Evidence	18
9.Front Display Page of B-CoC	19
10. Creation of Evidence	19
11. Evidence Submitted	20
12. Transfer and Get Evidence Interface	20
13. Remove Evidence Interface	21
14. Data Entered in Get Evidence	21
15. Get Evidence Function	22
16. Transfer of Evidence	23
17. Evidence Tempering	24
18. Error Message Due to Tempering	24

19.Evidence Removal Failed	25
20. Evidence Removed	25
21.Empty Evidence	26

CHAPTER-1

INTRODUCTION

1.1)Overview

From the time of collection of digital evidence until submitting it in a court special care should be taken to protect it from tempering. The Chain of Custody(CoC) is a process that gives information about how evidences has been collected, validated ,tracked and protected on it's way to court.

The main requirement of CoC is that it must ensure data Integrity(Data Integrity means no third party would able to modify data), Traceability(The evidence must be traced from time of it's collection until it is destroyed).

We are proposing a Blockchain based architecture for CoC of Digital Evideneces A Blockchain is a distributive immutable ledger.It consist of several blocks get connected to each other. Each block consists of data,32-byte hash value of that data and hash value of previous block. In our B-CoC model the blocks contains the ID of the evidence as hash value and description of the evidence as data. This blockchain technology implements a decentralized system in a peer to peer network.

In order to add a new block in blockchain various consensus protocols are used such as Proof-of-Work(PoW) and Proof-of-Stake(PoS).In PoW in order to add a node the miner node would have to do heavy computation which requires lot of memory and time then only it can add new block. Then the block would be verified by different system before adding it to blockchain. The main drawback of PoW is that it requires huge demand of energy.

In PoS there are set of nodes known as validators which propose to add new block and voting is done on them. Validators put a stake on network (a particular amount of cryptocurrency) and then in future if they were found to be corrupt then they would lose the stake.

There is Proof-Of-Authority(PoA) in which individual identity is at stake. With PoA all the validators must have been authorized and they should be well known. The authorized user would have only permission to access the particular blocks or to perform any operations.

There would be smart contracts which we will run on Ethereum Virtual Machine(EVM). Every node executes a local EVM. Whenever we want to execute a function on a smart contract, the node executes the function on the EVM and stores it in the blockchain. In EVM each operation would use a virtual resource known as gas. Gas is used to perform operations and include the evidence in the blockchain.

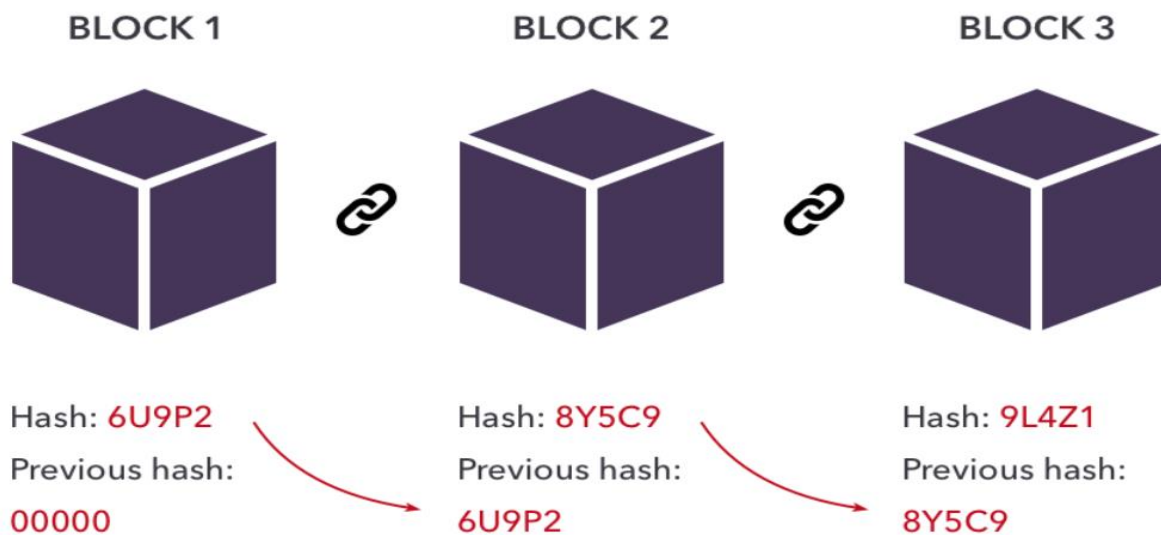


Fig:-1.1(Blockchain)

1.2) OBJECTIVES

The main objective of our project is that we ensure that evidence should not be tampered from the time of collection to time of presenting it in the court. In case if any evidence get tampered at any point then due to change in evidence, the hash value of evidence would be changed and we would come to know that evidence had been tampered and we would know about the person who had changed the evidence.

1.3)PROBLEM STATEMENT

In our work B-CoC(Blockchain Based Chain Of Custody) we have to take care that the digital evidence which consist of important information such as EvidenceID, description about crime should be collected and kept in a safe environment and care should be taken to ensure that evidence should not get tampered .

CHAPTER-2

LITERATURE SURVEY

Yudi Prayudi and Azhari SN in the year 2011[1] discussed the working of DEMF(Digital Evidence Management Framework). It include proper information about collection and preserving integrity of evidence.This ensure security to chain of custody based on (5W/1H). In first step we use SHA-2 to calculate fingerprint to generate a fixed-size value. In second step we will deal with Authentication. In third step we add a timestamp to know when evidence was created. In fourth step we use RFID(Radio Frequency identification tag) or GPS to track the movement of evidence. At the end we use asymmetric(private+public)key encryption.

Giuliano Giova and Brian[2] proposed the use of techniques such as Advanced Forensic Format(AFF) and Resource Description Framework(RDF).Advanced forensic format used to store image data along with metadata such as device identification,sector-size etc. RDX is a standard language that facilitates communication among different device belonging to different locations.

Miroslav Baca[3] in his work proposed the use of protege which is a software which provide understanding about digital forensic structure among software agents. It's main motive was to provide informations about different forensic equipments to users.

Silvia Bonomi, Marco Casini in the year 2018[4] proposed the idea about evidence Database in which we kept information about evidence (Evidence ID, Hash, Description) and in Evidence log we keep only important information such as Hash Of Evidence. Hash is generated by converting first image to binary character and then converting characters to Hash on the value of hashing algorithm such as SHA-256. In their paper there is B-coc architecture which comprises of different functions such as getEvidence, CreateEvidence, RemoveEvidence, TransferEvidence. getEvidence provide information about ID of evidence, hash of evidence, owner's name, description about evidence. CreateEvidence provide information about newly created ID of Evidence and its description along with owner name. TransferEvidence will give name of previous and current owner. RemoveEvidence help in removal of evidence.

Jasmin Cosic in the year 2017[5] proposed his work dealing with different important elements such as data integrity and confidentiality of the digital evidence. There are a number of issues regarding maintaining the digital evidence such as storing of digital evidences, maintaining metadata about evidence.

Meng Li, Mauro Conti in the year 2021[6] proposed of a lawful evidence management for digital forensic. Here the data first involves entire evidence flow from collection to examination, to analyse and report. LEChain achieves efficient management, guarantees control over flow of evidences from one user to another

Mark Scanion in the year 2019[7] made use of technique known as RAFT(Remote Acquisition Forensic Tools) to reduce the time taken to acquire the necessary evidence. The RAFT system is based on client/server architecture. In case of new client a new space created on server which stores all relevant files and information about clients.

Mrunali Chopade in the year 2019[8] proposed a work in which there is protection of evidences from a number of undesirable outcomes such as alteration or destruction. It includes all necessary steps which ensure that evidence has not been altered during times of its collection to usage in court.

David Billard in the year 2016[9] proposed a Digital Evidence Inventory(DEI). Here evidence is collected inside an immutable e-evidence and it provides traceability of the evidence.

.

CHAPTER-3

EXISTING WORK

There is Blockchain-Based chain of custody(B-CoC) architecture in which only the authorized and authenticated parties are allowed to view and manage digital evidences. B-CoC architecture is composed of three things such as

- i)EvidenceDB.
- ii)Evidence Log.
- iii) front end interface.

EvidenceDB is an ordinary database or file repository in which the original digital evidence is stored along with an ID. In order to obtain the ID of the digital evidence first the image is converted to binary format and then the binary code is converted to hash value using SHA-256 (32 byte) hash function generator.

Evidence Log is implemented through blockchain technology in which the ID of the evidence is stored in blocks along with different information such as complete history of the owners and the time according to which owner change. It contain information about current owner , previous owner . It ensure that when evidences transferred from one owner to another The ID(hash value) of evidence would be same(It would not be tempered).

The evidence log runs a smart contract which consist of four functions :-

- i)CreateEvidence(ID,Description).
- ii)TranferEvidence(ID,newowner).

iii)RemoveEvidence(ID).

iv)GetEvidence(ID).

Whenever a new evidence is discovered then after creation of ID of that evidence there would be a new evidence entry in evidence log with a specified ID, description, the person who discovered evidence would be set as initial owner. This all operations are performed by CreateEvidence function.

Then after a evidence is created it need to moved through a series of steps (police->forensic doctor->lawyer->judge). Then at each stage we need to transfer ownership of the evidence from current owner to new owner. For this we require a function known as TranferEvidence function.

Then at any point if we want to get informations about ID, description and different owners who have accessed the evidence in what order and now who is the current owner of the evidence then we require a function known as GetEvidence function in which we have to pass the Id of the evidence as input argument to the function.

Then afterward when the case get completed or if the evidence is found to be false (not of use) then the evidence would be removed from the block by the person who is the current owner of the block. Now there is a frontend. The frontend represent the interface between B-CoC and it's users. It show's the creation of different blocks and which block contain what type of EvidenceID along with the description of the evidence.

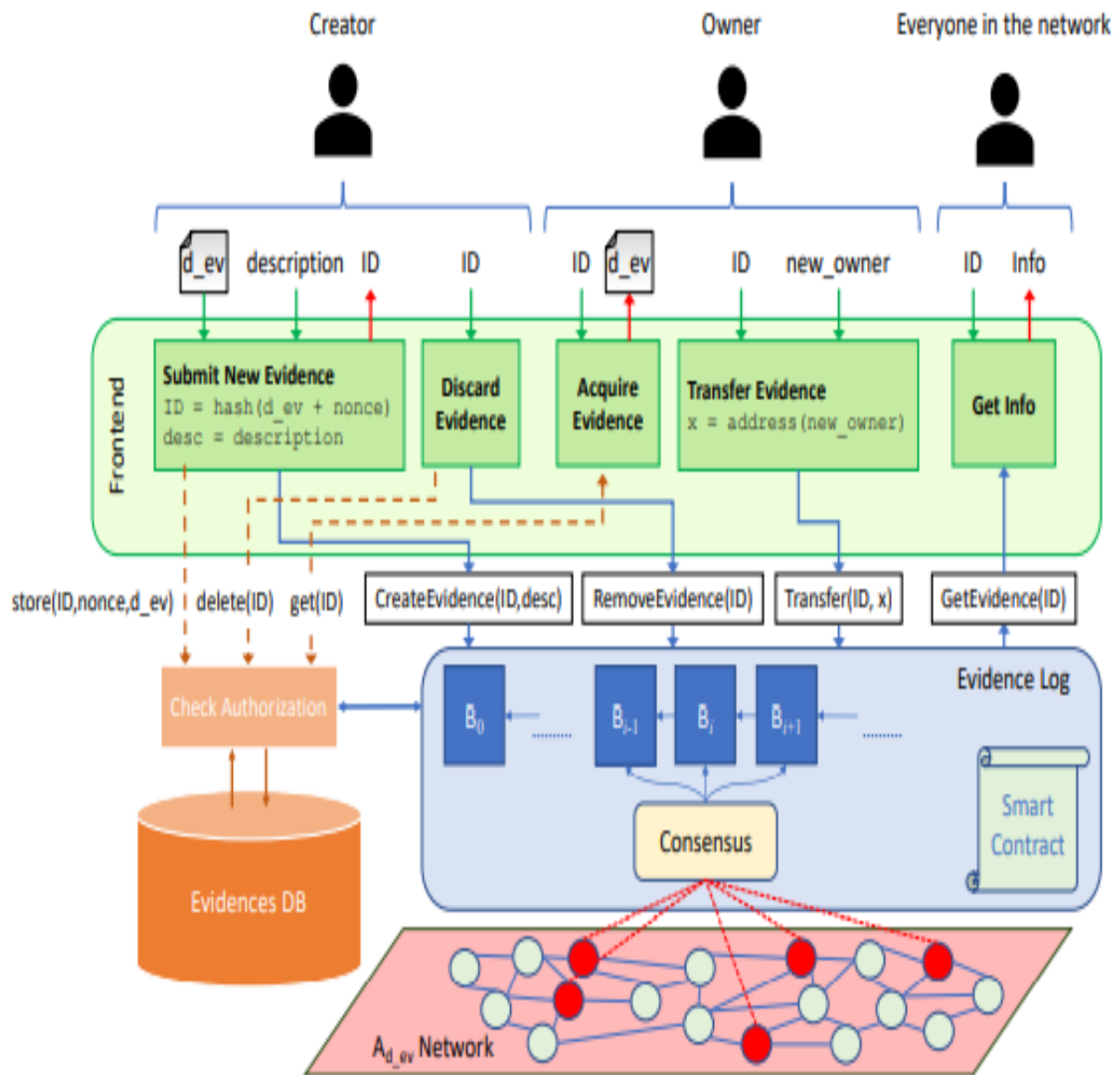


Fig:-2.1(B-CoC Architecture)

CHAPTER-4

PROPOSED WORK

4.1)OVERVIEW

Here we will deploy our blockchain on public network. The blocks of blockchain consist information about EvidenceID, Description about the evidence. In our smart contract we have deployed four functions namely GetEvidence, TransferEvidence, RemoveEvidence and CreateEvidence.

Whenever we get any image of evidence we have to generate evidenceID. The evidenceID is the hash value of 32 byte which we generate by the help of NodeJS cryptomodule. Then after getting the evidenceID from the image of evidence. We will first run our function CreateEvidence which takes evidenceID and description as initial input parameter. In CreateEvidence first step is to check that evidenceID should be a valid 32 byte value then the person who discovered evidence would be the initial creator of the block and there would be a time value generated when the evidence is created.

Then after evidence is created it need to be transferred to different person based on requirement. while transferring evidence first step is to check that proper evidence exist. Then if evidence is found to be true then it would be transferred to a new person .The previous owner of the block would be replaced by new owner. It include the time at which the transfer of ownership occurred.

In order to get information about EvidenceID, Evidence description, file Names, about the name of the owners and what time who is the owner of block. We get this all informations by the help of getEvidence function which takes ID of the Evidence as the input argument.

Then at last when the case get over or the evidence is found to be false we will remove the evidence from the block for this we use function removeEvidence.

Here we created the interface for the functions by the help of EJS template engine by the help of which we can send request in order to run our functions.

In our work if any person would change the evidence by changing the image of the evidence by a new image then we can detect that there is a change in evidence and we can know about the person who had tempered the evidence.

In this work we dealt with multiple evidences related to a crime and we used cumulative hash technique to generate a unique hash by combining hash of all evidences. So if there would be tempering on any evidence there would be change in cumulative hash, Hence by this we come to know that evidence had been modified by which user. By doing this data-integrity of evidence is preserved.

4.2)ALGORITHM

1. Evidence Id(32 byte hash value) will be generated by SHA-256 Algorithm.
2. After the creation of new evidence, the CreateEvidence(ID,description) function is called by help of which evidenceID ,description and initial owner name would be stored in blocks.
3. Check if evidenceID is valid then perform steps:-
 - i)Evidence[ID].ID=ID.
 - ii)Evidence[ID].description=description.
 - iii)Evidence[ID].owner=ownername.
 - iv)Evidence[ID].valid=1.
- 4.Then for transfer of evidence first check that evidenceID!=NULL ,it should be valid 32 byte.If valid then perform steps:-
 - i)Evidence[ID].owner=newowner .
 - ii)Evidence[ID].ttime.push(now).
 - iii)Evidence[ID].taddr.push(newowner).
- 6.Then if on performing getEvidence(ID) we get information such as:-
Evidence[ID].ID.
Evidence[ID].owner.
Evidence[ID].creator.
Evidence[ID].ttime.
Evidences [ID]. evidenceArray
- 7.Finally there would be termination or deletion of evidence on completion of case by help of remove evidence[ID]

4.3) FLOWCHART

1)Creation of Evidence

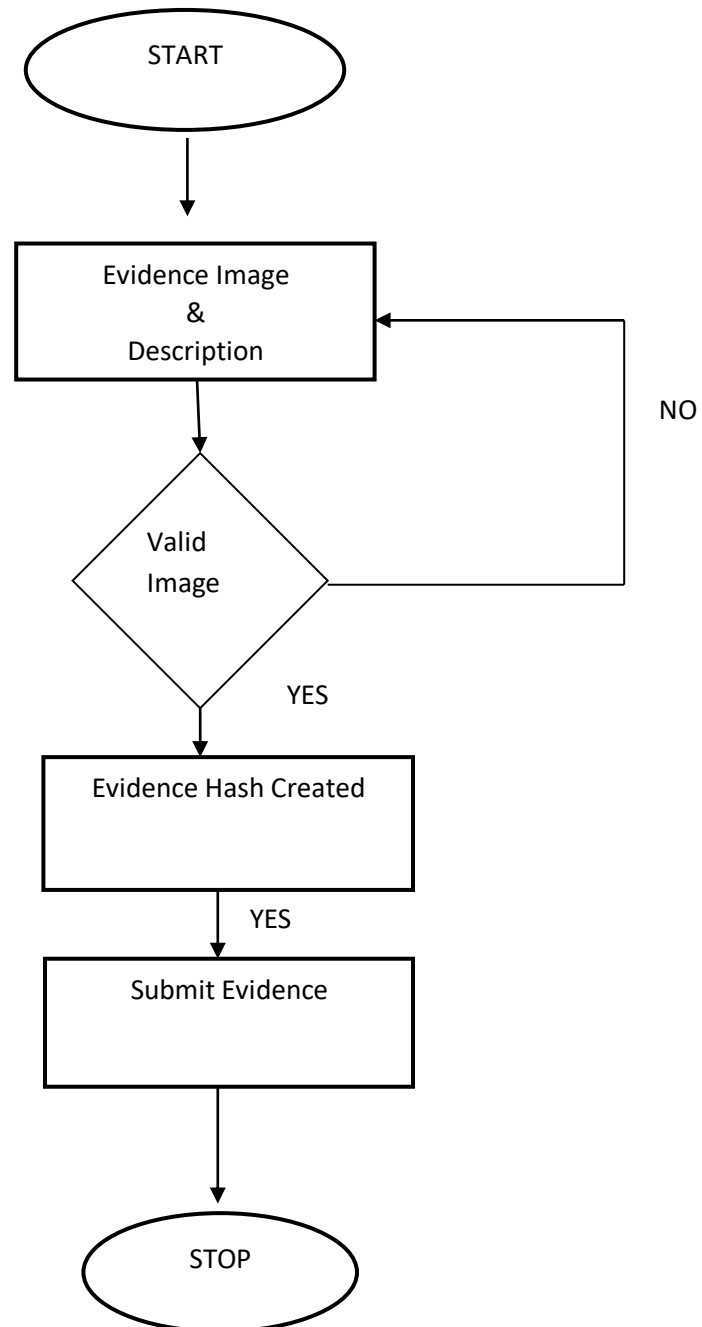


Fig:-4.1 Evidence creation

2)Transferring Of Evidence

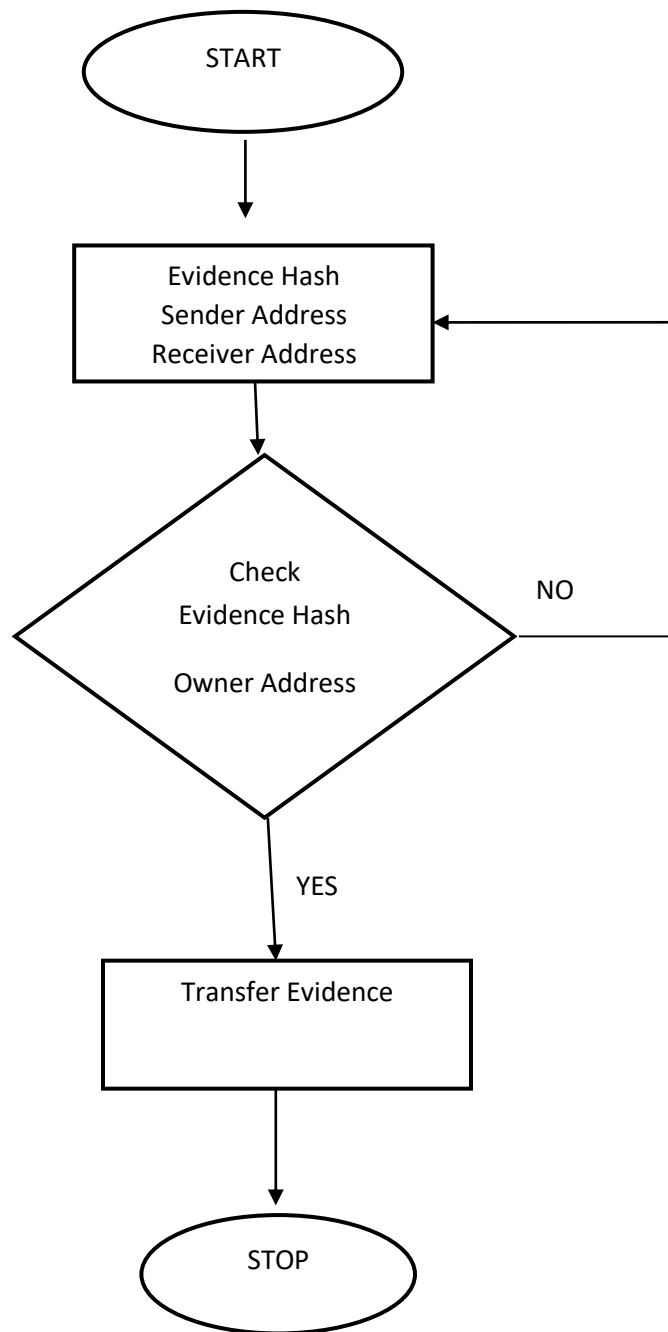


Fig:-4.2 Evidence Transfer

3)Evidence Display

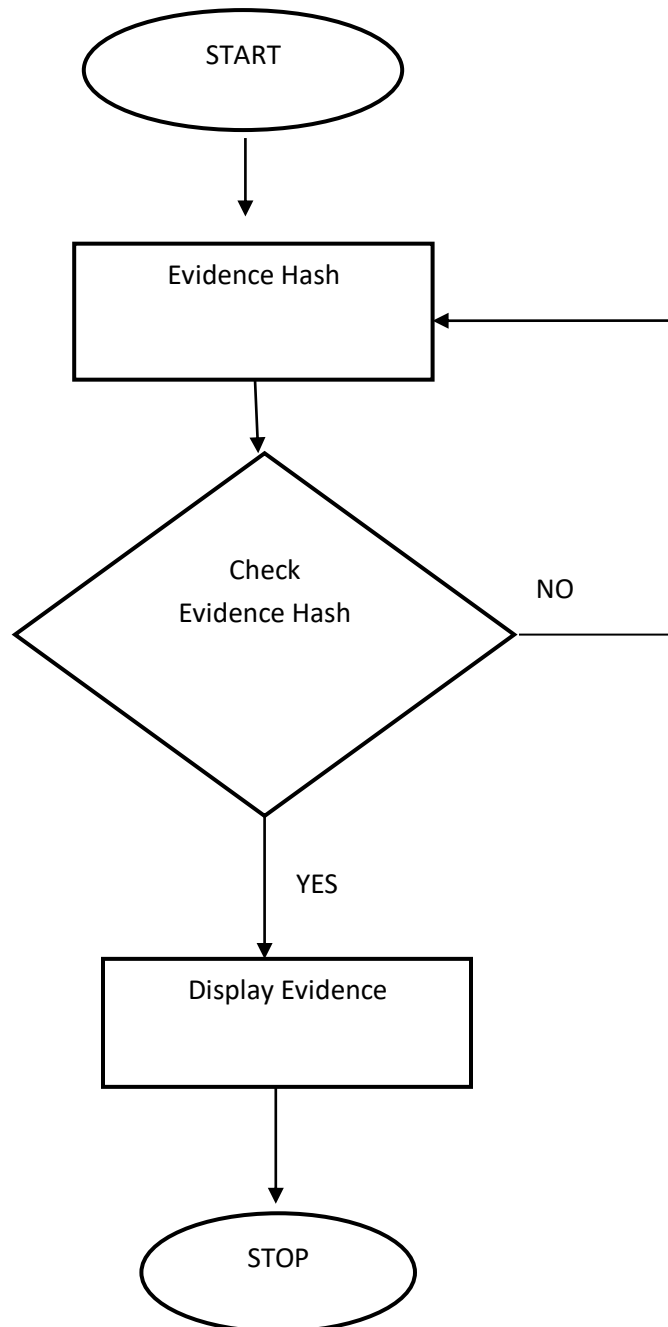


Fig:-4.3 Evidence Display

4)Removal of Evidence

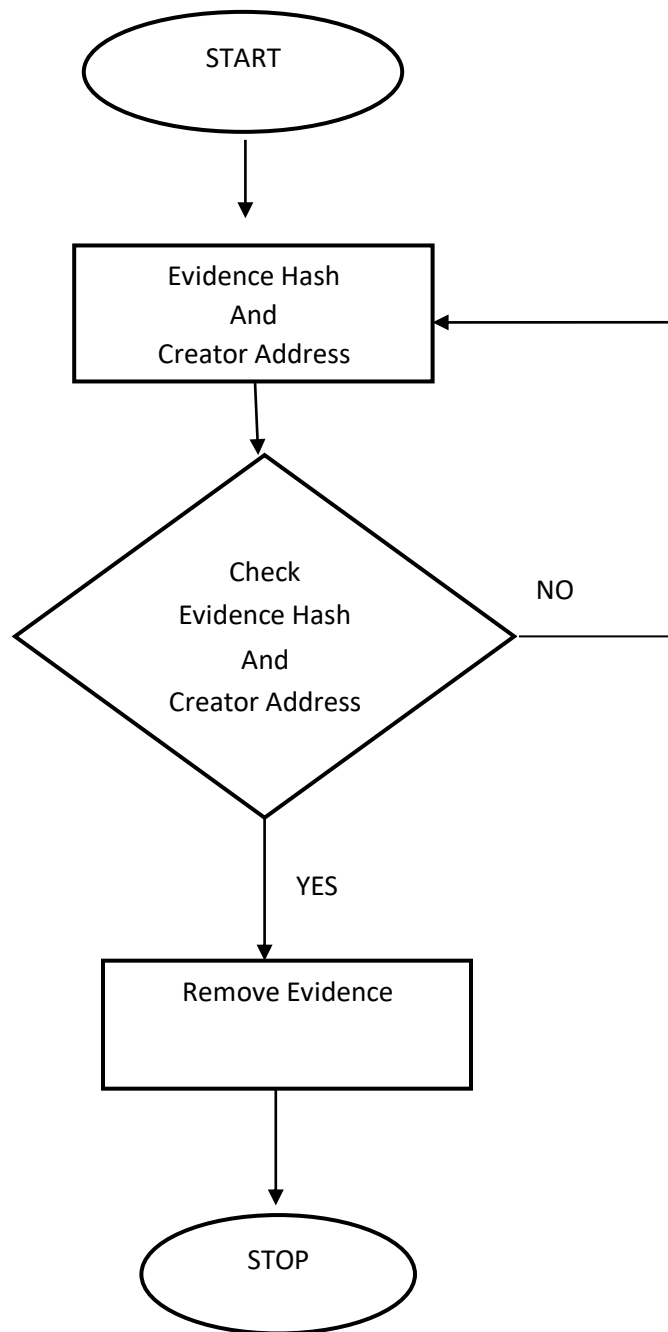


Fig:-4.4 Evidence Remove

5) Procedure For generating Hash

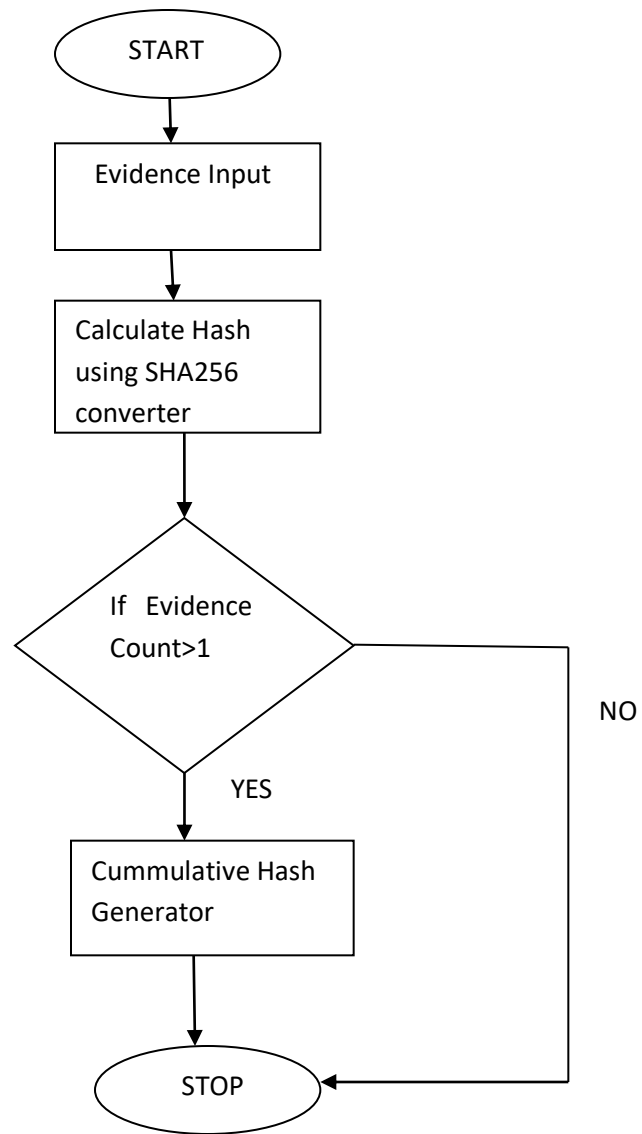


Fig:-4.5 Hash Generator

6) Detecting error due to change in evidence

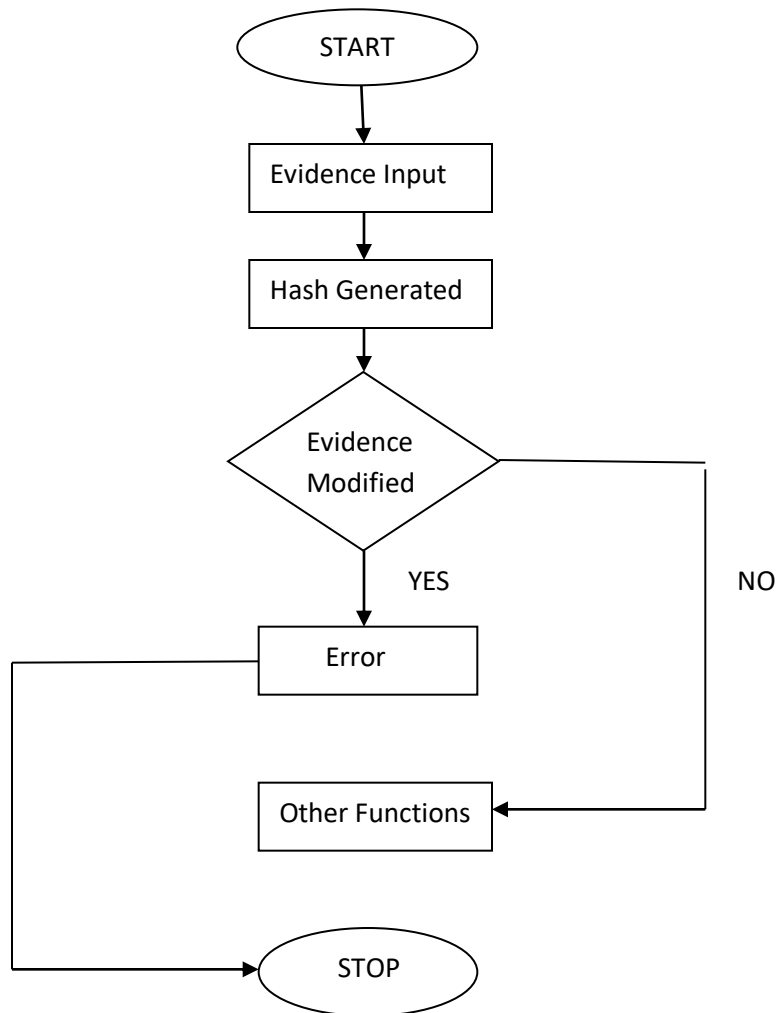


Fig:-4.6 Detection Of error due to change in Evidence

CHAPTER-5

RESULTS AND DISCUSSION.

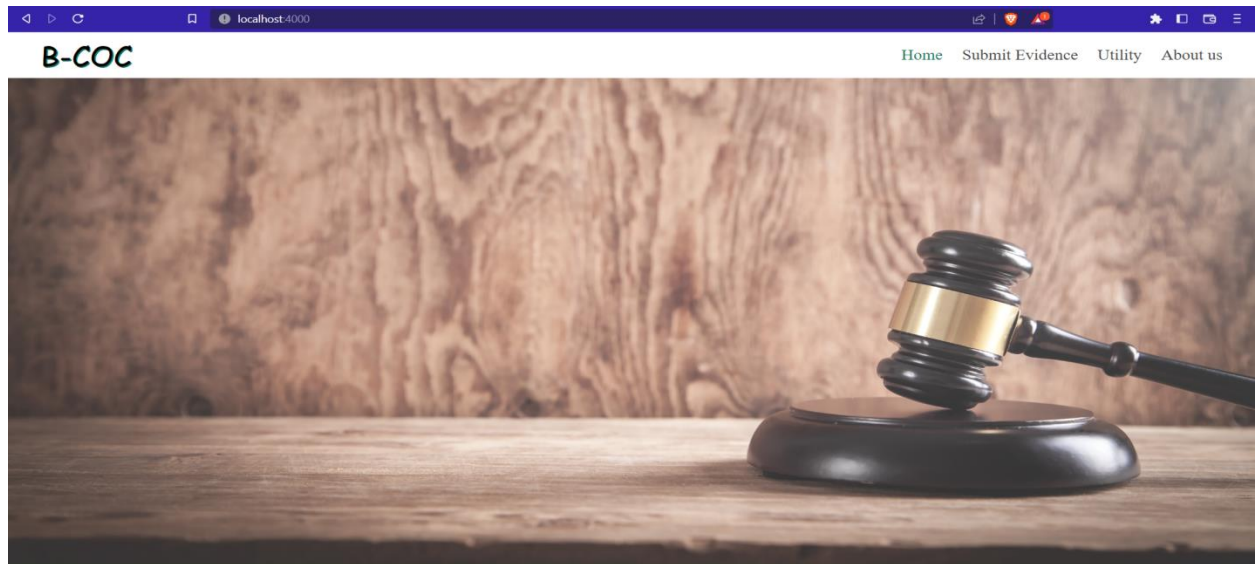


Fig:-5.1(Front Display Page of B-CoC)

This is the home page of Evidence project.

A screenshot of the 'Submit The Evidences' form on the B-COC website. The form is set against a light green background. It contains the following fields and elements: a title 'Submit The Evidences', a text input for 'Enter your Ethereum network address:' with a sample address '0xC075d113BfT1078D75b67982wFA575t6ATQ230', a text input for 'Description for Crime:' with the word 'Murder' entered, a file upload section for 'Evidence Images:' with a 'Choose Files' button and a '2 files' indicator, and a green 'submit' button at the bottom.

Fig:-5.2(Creation Of Evidence)

Here evidence is created by loading evidence image along with description of evidence and initial owner address.

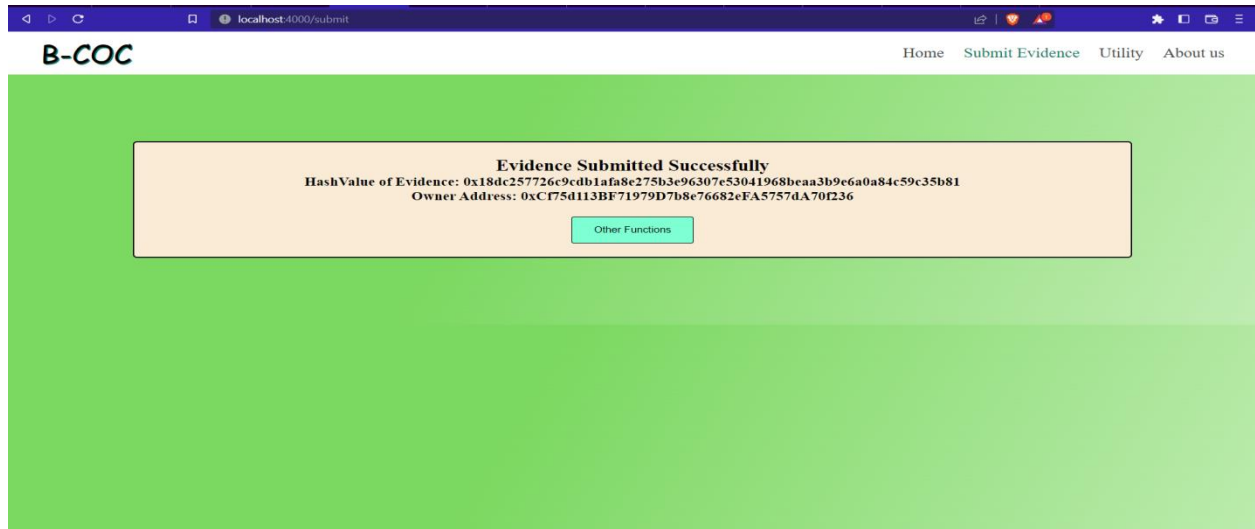


Fig:-5.3(Evidence Submitted)

After successful submission of evidence this interface is formed.

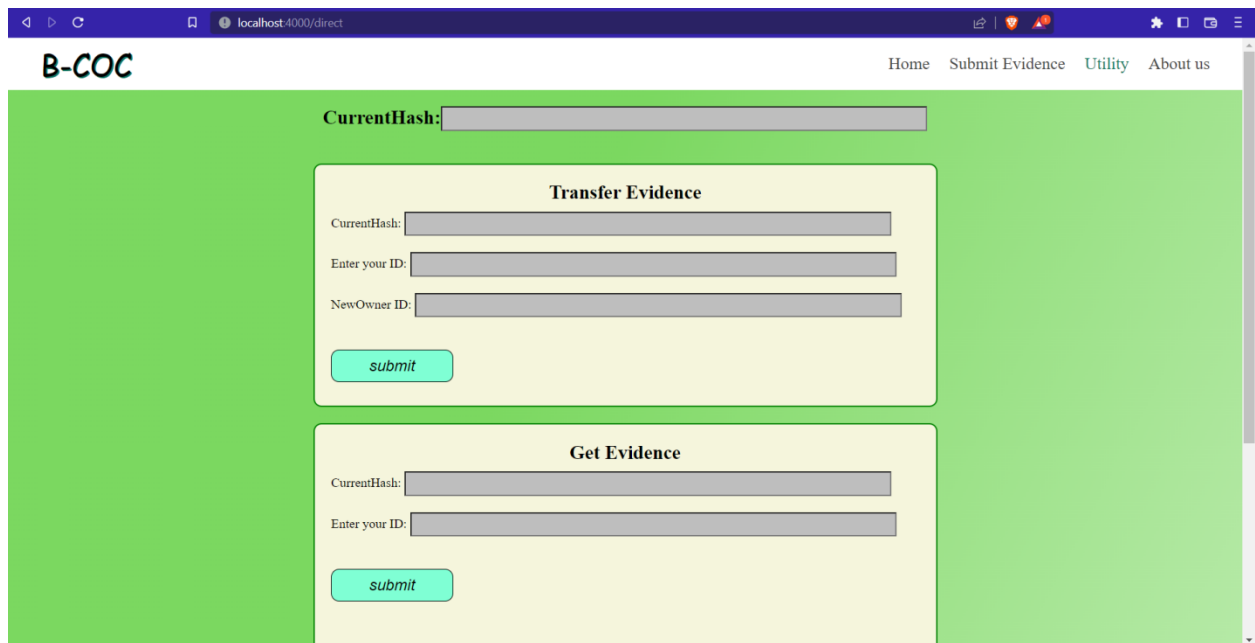


Fig:-5.4(Transfer And Get Evidence Interface)

This is the interface of Transfer Evidence function and Get Evidence function.

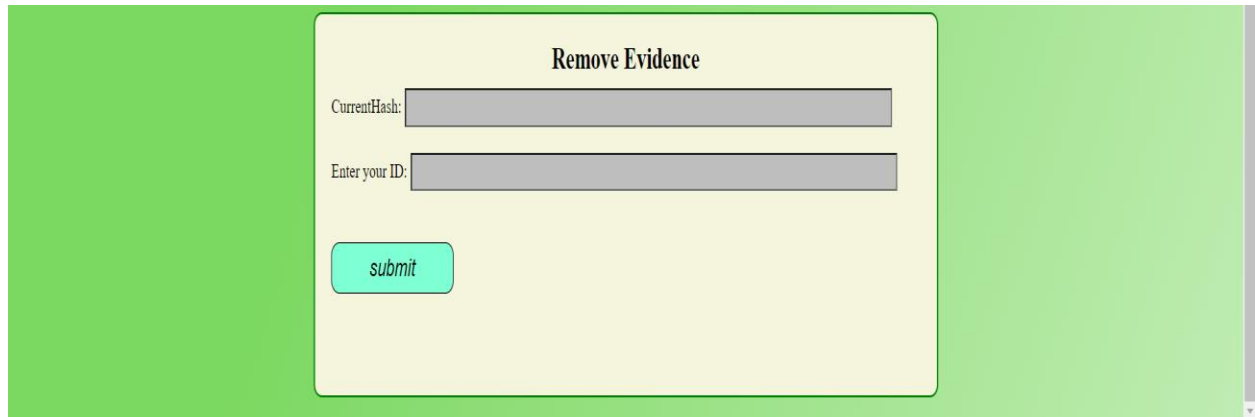
A screenshot of a web interface titled "Remove Evidence". It features a light yellow background with a green border. At the top, the title "Remove Evidence" is centered. Below the title, there are two input fields: "CurrentHash:" and "Enter your ID:". Both fields are currently empty. Below these fields is a green button with the text "submit" in a light blue font.

Fig:-5.5(Remove Evidence Interface)

Here we remove the submitted Evidence.

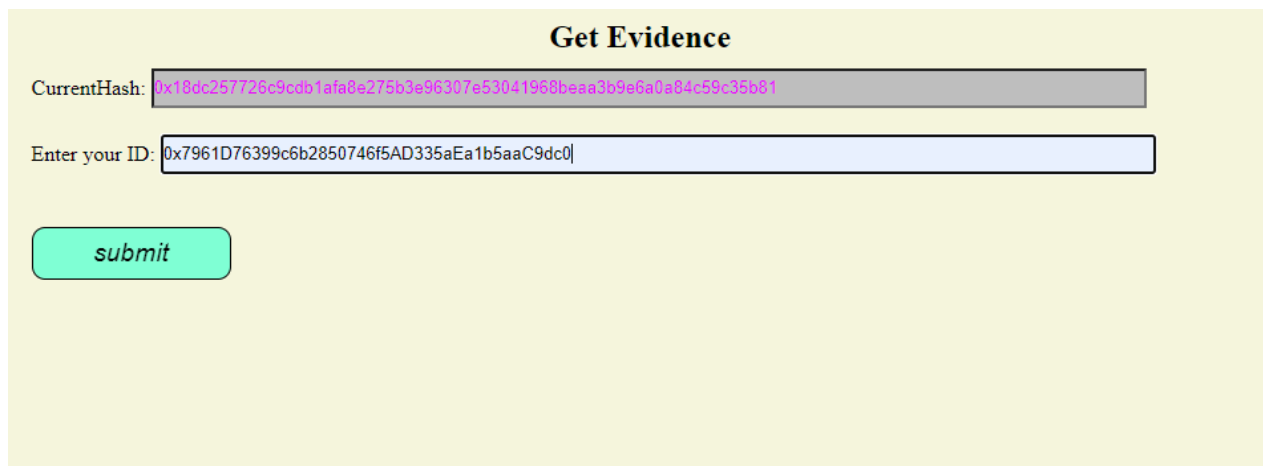
A screenshot of a web interface titled "Get Evidence". It features a light yellow background. At the top, the title "Get Evidence" is centered. Below the title, there are two input fields: "CurrentHash:" and "Enter your ID:". The "CurrentHash:" field contains a long hexadecimal string: 0x18dc257726c9cdb1afa8e275b3e96307e53041968beaa3b9e6a0a84c59c35b81. The "Enter your ID:" field contains a long hexadecimal string: 0x7961D76399c6b2850746f5AD335aEa1b5aaC9dc0|. Below these fields is a green button with the text "submit" in a light blue font.

Fig:-5.6(Data Entered in Get Evidence)

It is an example of the data entered in Get Evidence Function.



Fig:-5.7(Get Evidence Function)

This is the output of the Get Evidence Function . It provide various details about evidence file name,owner of evidence.

localhost:4000/transferEvent

B-COC

Home Submit Evidence Utility About us

Evidences Transfer Details

Hash Of New Evidence:

1x18c257726c9cd1afa6a275a3e96307e53041960aaa3b9ef6a04c59c35d1

Hash Of Old Evidence:

1x18c257726c9cd1afa6a275a3e96307e53041960aaa3b9ef6a04c59c35d1

Owner Of The Evidence:

1x6780484244c7f04a1139000787a042c13248E7

Receiver Of The Evidence:

1x075a1139f7197d78a768024f457570470238

Description Of Transference:

The Data has been Transferred Successfully

Evidence File Name:

pattern0104.jpg

pattern0102.jpg

Go to home

Fig:-5.8(Transfer Of Evidence)

It is the working of the Transfer Evidence function. Evidence transfer from initial owner to new owner.

B-COC

[Home](#) [Submit Evidence](#) [Utility](#) [About us](#)

Evidences Transfer Details

Hash Of New Evidence:

0xc59967897634f99b0714864704023ca8015e10638856ac45db03a4e44316b35d2

Hash Of Old Evidence:

0x184c257726c3c0b1afa0e275b3496307a53041968b0aa3b9e6a0a04c58c35a01

Owner Of The Evidence:

0xcd75d113b071979d7b8a76682efa5757da70d236

Receiver Of The Evidence:

0x57b04b420fc0f04a1139800d707a842c132abe7

Description Of Transference:

The Data has been Changed When the Owner was : 0xcd75d113b071979d7b8a76682efa5757da70d236

Evidence File Name:

patterns0304.jpg

patterns0102.jpg

Go to home

Fig:-5.9(Evidence Tempering)

When the evidence is tempered due to which the hash value change, this interface is shown.

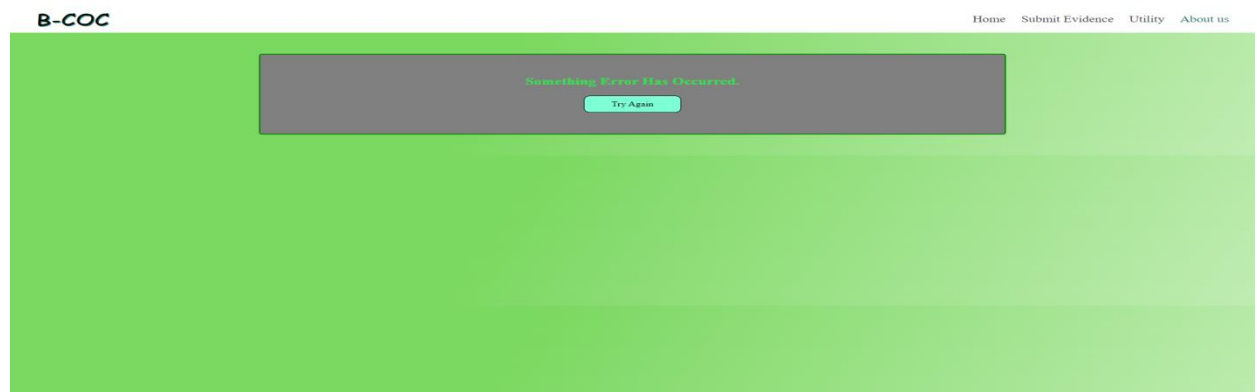


Fig:-5.10(Error Message due to tempering)

This error message is displayed due to tempering of evidence.

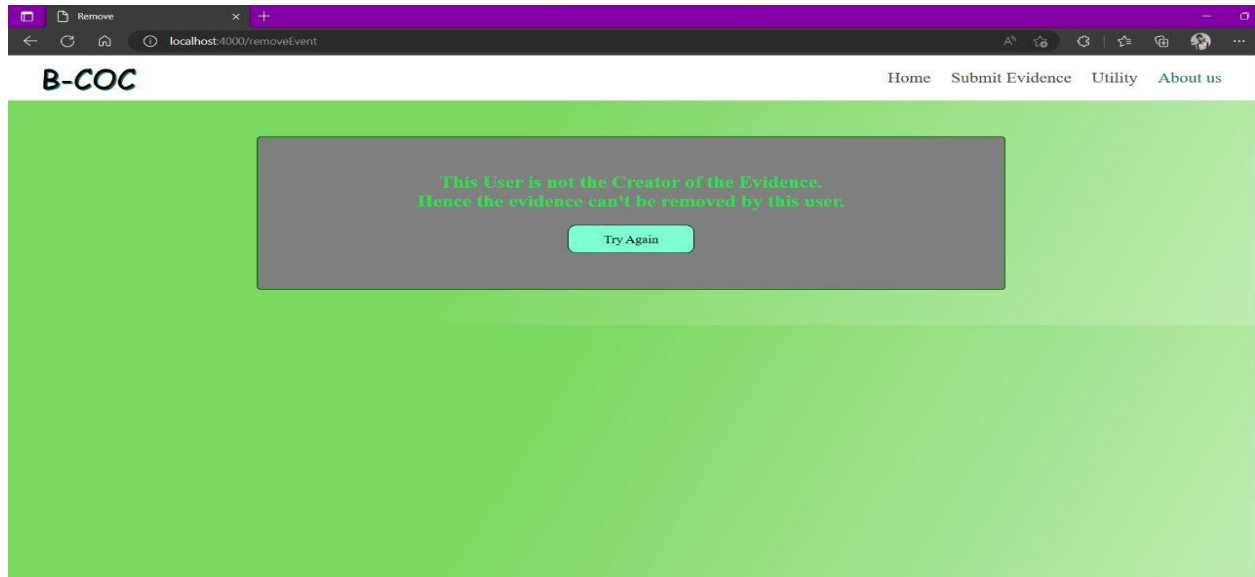


Fig:-5.11(Evidence Removal Failed).

This interface is the display of message that evidence can't be removed by any other user other than creator.

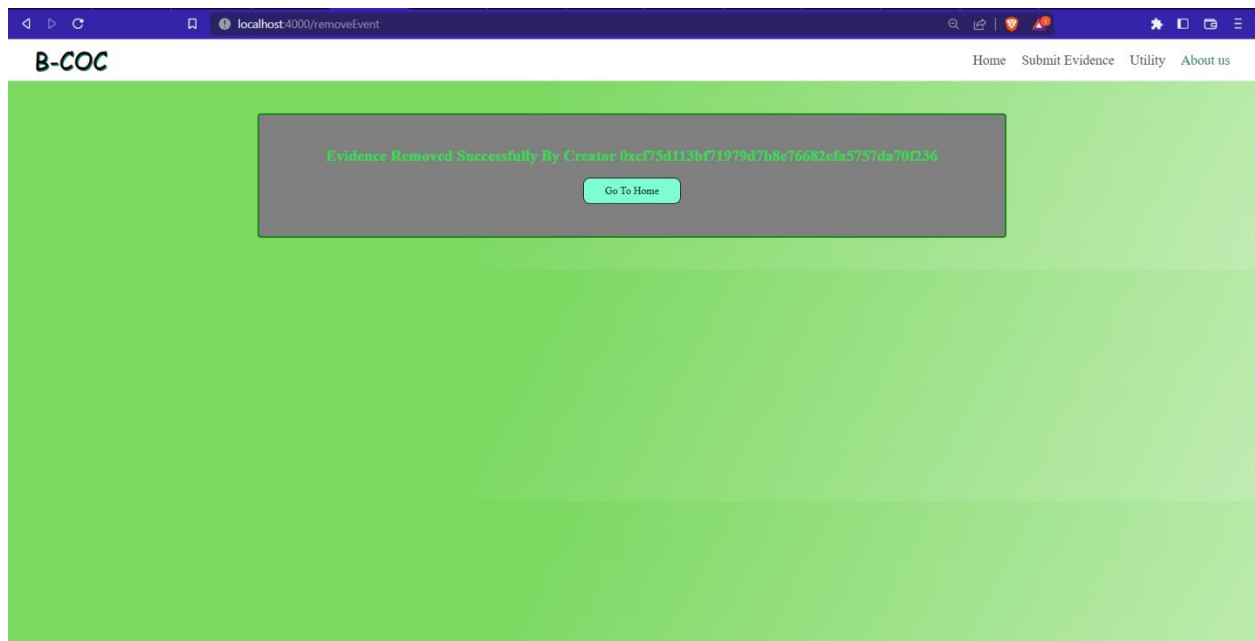


Fig:-5.12(Evidence Removed).

After successful removal of evidence by the creator this interface is shown.

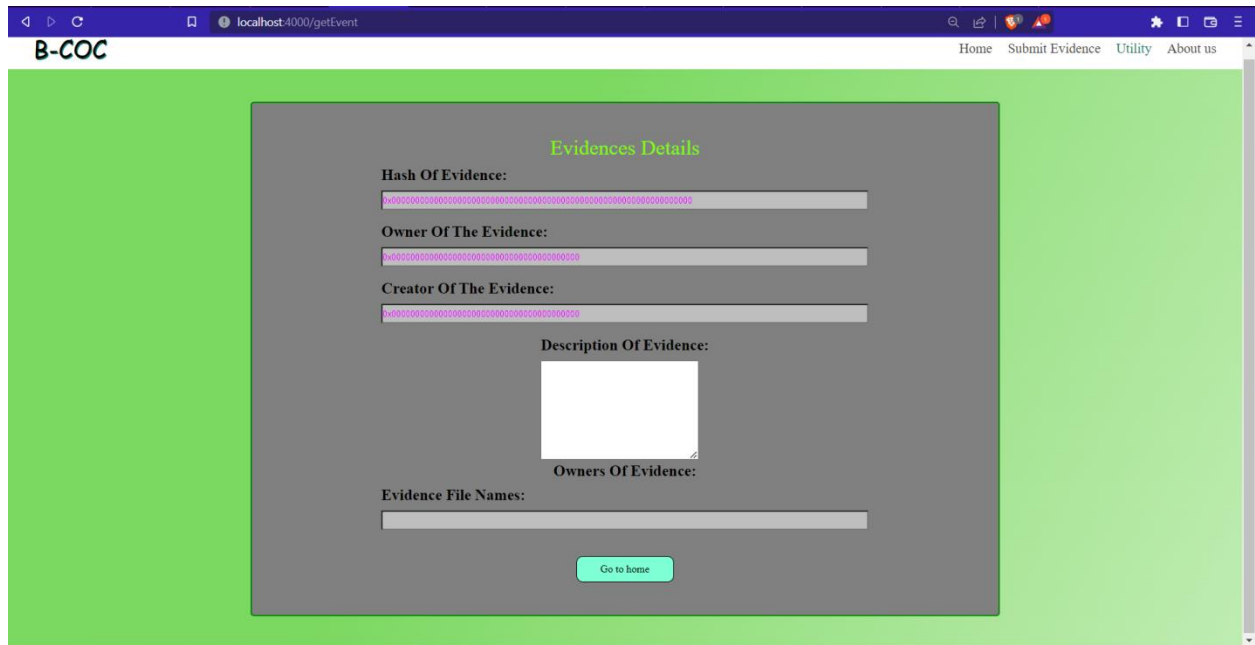


Fig:-5.13(Empty Evidence)

This interface is shown when the evidence had been successfully removed so it's hash value and description and owner name set to NULL.

CHAPTER-6

CONCLUSION AND FUTURE SCOPE

We represented B-CoC Architecture to show the different process need to be followed in order to deal with evidence. In this paper we implemented different types of function right from collection of evidence to display the description ,ID and owner name of the evidence. In this paper we ensure that when evidence passed from person to person and the evidence would not be tempered. In this we dealt with different methods that can temper our evidence such as evidence cropping, evidence change, evidence resolution. One of the limitation is that in case of multiple evidences on tempering of any evidence we come to know that evidence had been tempered but we can't say which particular evidence was modified and by which technique the evidence had been modified

REFERENCES

- [1] Giova, G. (2011). Improving chain of custody in forensic investigation of electronic digital systems. *International Journal of Computer Science and Network Security*, 11(1), 1-9.
- [2] Carrier B. Defining digital forensic examination and analysis tools using abstraction layers. *International Journal of digital evidence*. 2003 Jan;1(4):1-2.
- [3] Brinson, Ashley, Abigail Robinson, and Marcus Rogers. "A cyber forensics ontology: Creating a new approach to studying cyber forensics." *digital investigation* 3 (2006): 37-43.
- [4] Castro, Miguel, and Barbara Liskov. "Practical byzantine fault tolerance." In *OsDI*, vol. 99, no. 1999, pp. 173-186. 1999.
- [5] Ćosić, Jasmin, and Miroslav Bača. "(Im) proving chain of custody and digital evidence integrity with time stamp." In *The 33rd International Convention MIPRO*, pp. 1226-1230. IEEE, 2010.
- [6] Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y. and Muralidharan, S., 2018, April. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the thirteenth EuroSys conference* (pp. 1-15).
- [7] Vasin, Pavel. "Blackcoin's proof-of-stake protocol v2." URL: <https://blackcoin.co/blackcoin-pos-protocol-v2-whitepaper.pdf> 71 (2014).
- [8] Flores, Denys A., and Arshad Jhumka. "Implementing chain of custody requirements in database audit records for forensic purposes." In *2017 IEEE Trustcom/BigDataSE/ICSS*, pp. 675-682. IEEE, 2017.
- [9] Sadiku, Matthew NO, Adebawale E. Shadare, and Sarhan M. Musa. "Digital chain of custody." *Int. J. Adv. Res. Comput. Sci. Softw. Eng* 7, no. 7 (2017): 117.
- [10] J. Ćosić, Z. Ćosić, M. Bača, J. Cosic, G. Cosic, and M. Baca, "An Ontological Approach to Study and Manage Digital Chain of Custody of Digital Evidence," *JIOS*, vol. 35, no. 1, pp. 1–13, 2018

- [11] Bentov, I., Gabizon, A., Mizrahi, A.: Cryptocurrencies without proof of work. In: Financial Cryptography Workshops. Lecture Notes in Computer Science, vol. 9604, pp. 142–157. Springer (2019)
- [12] Giova, G., 2011. Improving chain of custody in forensic investigation of electronic digital systems. International Journal of Computer Science and Network Security, 11(1), pp.1-9.
- [13] CARRIER, B. D.; SPAFFORD, E. H. An Event-Based Digital Forensic Investigation Framework. Digital Forensic Research Workshop 2018. Baltimore, Maryland: [s.n.]. 2018.
- [14] COMMON DIGITAL EVIDENCE STORAGE FORMAT WORKING GROUP. Survey of Disk Image Storage Formats. Digital Forensic Research Workshop (DFRWS). [S.l.], p. 1-18. 2006
- [15] NOBLETT, M. G.; POLLITT, M. M.; PRESLEY, L. A. Recovering and Examining Computer Forensic Evidence. Forensic Science Communications, v. 2, n. 4, October 2000.
- [16] DIGITAL FORENSIC RESEARCH WORKSHOP (DFRWS). A Road Map for Digital Forensic Research. Report From the First Digital Forensic Research Workshop (DFRWS). Utica, New York: [s.n.]. 2001.
- [17] REITH, M.; CARR, C.; GUNSCH, G. An Examination of Digital Forensic Models. International Journal of Digital Evidence, 2002.
- [18] CARRIER, B.; SPAFFORD, E. H. Getting Physical with the Digital Investigation Process. International Journal of Digital Evidence, 2003.
- [19] Miguel Castro and Barbara Liskov. Practical Byzantine Fault Tolerance. In Proceedings of the Third Symposium on Operating Systems Design and Implementation, OSDI '99, pages 173–186, Berkeley, CA, USA, 1999. USENIX Association. URL: <http://dl.acm.org/citation.cfm?id=296806.296824>.
- [20] S. L. Garfinkel, “Digital forensics research: The next 10 years,” Digit. Investig., vol. 7, pp. S64–S73, Aug. 2017.

Major report(1).docx

ORIGINALITY REPORT

8%

SIMILARITY INDEX

PRIMARY SOURCES

- 1

deepai.org
Internet

157 words — 5%
- 2

"Digital Forensics and Cyber Crime", Springer Science and Business Media LLC, 2010
Crossref

17 words — 1%
- 3

modex.tech
Internet

15 words — 1%
- 4

Meng Li, Chhagan Lal, Mauro Conti, Donghui Hu. "LEChain: A blockchain-based lawful evidence management scheme for digital forensics", Future Generation Computer Systems, 2020
Crossref

13 words — < 1%
- 5

Scanlon, Mark. "Enabling the Remote Acquisition of Digital Forensic Evidence Through Secure Data Transmission and Verification.", University College Dublin (Ireland), 2018
ProQuest

11 words — < 1%
- 6

David Billard. "Weighted Forensics Evidence Using Blockchain", Proceedings of the 2018 International Conference on Computing and Data Engineering - ICCDE 2018, 2018
Crossref

8 words — < 1%
- 7

link.springer.com

8

Mauro Conti, Sandeep Kumar E, Chhagan Lal, Sushmita Ruj. "A Survey on Security and Privacy Issues of Bitcoin", IEEE Communications Surveys & Tutorials, 2018

Crossref

7 words — < 1%

EXCLUDE QUOTES	OFF	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF