Securing Logs of a System - An IoTA Tangle Use Case

Mohan Bhandary
Department of Electronics and
Telecommunication Engineering
Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Andheri (W), Mumbai, India
mohan.bhandary@spit.ac.in

Manish Parmar
Department of Electronics and
Telecommunication Engineering
Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Andheri (W), Mumbai, India
manparmar@yahoo.in

Dayanand Ambawade
Department of Electronics and
Telecommunication Engineering
Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Andheri (W), Mumbai, India
dd_ambawade@spit.ac.in

Abstract—**Outbreak of the Internet and technological dependence make the Cyber Technology Industry grow at a very fast pace. Cyber-security is the dire need of today which helps prevent or defend the systems from cyber-attacks. Distributed Ledger Technology like Blockchain with its various properties and advantages opens many opportunities for its use in thecyber-security domain and mitigating the associated risks. These technologies remove the need of any central trusted party or authority, which would facilitate recording and tracking of resources. This paper focuses on the use of Distributed Ledger Technology (DLT) in the field of Cyber-security. It focuses on the discussion of new distributed ledger technology IOTA, its features and working. This paper also focuses on one of the uses of the IOTA Tangle technology, which provides a solution for securing logs of a system. As the logs of a system are the most important evidence in the investigation phase of Cyber-Forensics, thus preserving the logs securely and maintaining their integrity is mandatory. Later in this paper, the use of IOTA is demonstrated for securing the logs of a system.**

Index Terms—**Distributed Ledger Technology (DLT), Blockchain, IOTA, Tangle, Masked Authenticated Messaging(MAM), Digital Forensics, Cybersecurity, System Logs.**

I. INTRODUCTION

Even after twelve years of publication of the white paper on Bitcoin in 2008 by Satoshi Nakamoto, the technology behind Bitcoin i.e. The Blockchain continues to gain popularity. With its unique properties, Blockchain Technology has emerged as a disruptive technology in various domains, like finance, supply chain management, smart contracts, healthcare, Energy, IoT, security and many other. [3]–[5] Recently, the number of research on Blockchain Technology has tremendously increased by people ranging from academic researchers, developer, industry experts, who are exploring the use of Blockchain Technology in various domains as mentioned above. A Blockchain is a secure, tamper-proof, shared and distributed ledger that is capable of recording and tracking resources without the need of a centralized trusted authority. Blockchain can be thought of a decentralized data management platform that provides immutability. Blockchain facilitates communication between two parties along-with resource exchanging within a peer-to-peer network where decisions are taken by the majority of the distributed peers instead of a single authority. It is provably secure against an adversary who tries to gain control of the system by attacking and compromising the centralized entity in a network. There are generally two categories of Blockchains: permissionless and permissioned. A permissionless blockchain (e.g., Ethereum, Bitcoin) is open to the public, and every transaction is to be validated by every or majority of participants. While only the authenticated users can join a permissioned blockchain. Currently, the potential of this disruptive technology has been proved in an application which requires a centralized entity. Among the Blockchain's promising applications are network monitoring and security services including authentication, confidentiality, privacy, integrity, and provenance.

Digital forensics or computer forensics is a type of forensics study that involves identifying, preserving, collecting, analyzing and reporting of any evidence available on digital media such as a computer, mobile, storage device, etc. According to a definition provided by NIST [17], computer forensics or Digital forensics is an applied science to identify an incident, collection, examination, and analysis of evidence data. While performing forensic investigation, the integrity of the information is to be preserved and also the chain of custody is to be followed strictly. Any Forensics process starts after the incident happens, as a post-incident activity. It follows through pre-defined steps, shown in Fig. 1.

Logs are a well maintained and documented record of events happening in the system with accurate time-stamp, which makes it very useful evidence while taking legal

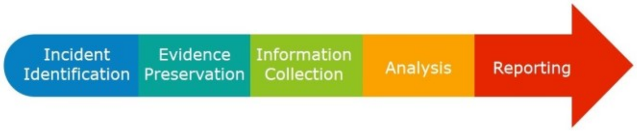


Fig. 1. Steps of Digital forensics

action against culprit or suspect. Even though logs are a very crucial part/ evidence in any digital investigation, collection and preservation logs are difficult. Its the responsibility of an individual or an organization to protect their sensitive data, which is a very crucial resource in important business decisions. The user who accesses a system and utilizes these systems leave some digital traces or their actions are been recorded in the Log files of the system. Later in the occurrence of an incident, the investigator or auditor will analyze these files for any important clue or evidence. Till now there is not any technique to can ensure the preservation oflogs as well as the integrity verification of logs. Consumers and forensic investigators (FI) trust on the service provider (SP) or any remote logging service to get appropriate activities log. Even though the logs are an essential element in the process of digital forensic investigation, the reliability of logs is still cannot be a guarantee as an attacker/ hacker can easily change/add/rearrange/delete the logs of the compromised system, to avoid been detected and remain untraceable. On the other hand, the forensic investigator (FI) also can be malicious and can modify/add/delete/reorder the logs before presenting in the court.

In this paper, the distributed ledger technology is used, for preserving the integrity of the logs such that they can remain tamper-proof. As discussed above regarding the features of the Blockchain, which make it disruptive technology, out of which few are decentralization, tracking and tracing, confidentiality, fraud security, sustainability, integrity, smart contracts, availability. Few research papers described in the next section shows how Blockchain has been used for the work of preserving the logs. In this paper, focusing on using IOTA, a new distributed ledger technology for preserving the integrity of the logs. In later sections, IOTA is discussed in detail and discuss our implementation using IOTA.

II. Related Works

The research papers enlisted below give an idea of how the Blockchain concept is applied to secure forensically important logs.

The authors in the paper BlockSLaaS: Blockchain Assisted Secure Logging-as-a-Service for Cloud Forensics [6], described the use of Blockchain for providing secure logging service mainly focused on the cloud environment. They explained the various threats on the forensically important logs along with their implementation on a Blockchain platform called Hyperledger fabric, for preserving the logs and verifying its integrity and confidentiality.

Also, a research paper titled “JusticeChain: Using Blockchain to Protect Justice Logs” [7] by R. Belchior, M. Correia and A. Vasconcelos also focus on the similar objective of securing logs of the system and preserving these log to facilitate their use in legal or judicial systems. The system described in the paper above is implemented in Portugal judicial system based on Blockchain for the secure storage of the data/ digital evidence. The JusticeChain system consists of mainly two Blockchain components namely the Blockchain and the Blockchain Client Component. The Blockchain client component is used for saving the important logs of the system and it allows user to access the Blockchain where all these saved data is stored.

III. IOTA Technology Overview

With the rising popularity of Bitcoin after 2008, the technology behind it i.e. the Blockchain has been adopted in various domains. However, the Blockchain Technology (BCT) has few drawbacks which hinder it from been used as a standard platform in the IoT domain. Few noteworthy drawbacks of Blockchain are the scalability issue, the use of transaction fees, transaction approval rate, no provision for sending small money transactions, and other. These drawbacks make blockchain not an appropriate technology for the machine to machine interactions and Internet of Things (IoT) economy. [8]

IOTA is open-source distributed ledger technology, launched in 2018, which makes it possible to transfer data or token between connected devices among each other with zero transaction fees. IOTA is a cryptocurrency for Internet of Thing (IoT) communication and machine to machine communication. IOTA aims to provide a solution to the key drawback of the Bitcoin system [2]. IOTA is a Greek word which means infinitesimally small, this is about the possibility of making micro-transaction with the help of IOTA.

There is conceptually no blocks and chain in IOTA, instead, it uses the Tangle technology, as shown in fig 2. The Tangle is a distributed ledger architecture based on the DAG (Directed Acyclic Graph) structure [1]. Also, there is no miner in IOTA, the transaction itself validates the previous two transactions, thus no transaction fees exist. Yet at the center, the Tangle works on a similar fundamental principle as that of the Blockchain which is: being a distributed ledger/ database, a P2P network, depends on a consensus and validation mechanism.

The IOTA has no global blockchain as a ledger, instead there exists a graph called Tangle, which acts as the ledger for storing all data / transactions. The IOTA Tangle can be considered as a continuous stream of individual transaction which is interlinked to each other. The Tangle consists of :

1) nodes:- entities that perform issuing and validation of transactions;
2) genesis transaction:- the first transaction which has all the IOTA token;

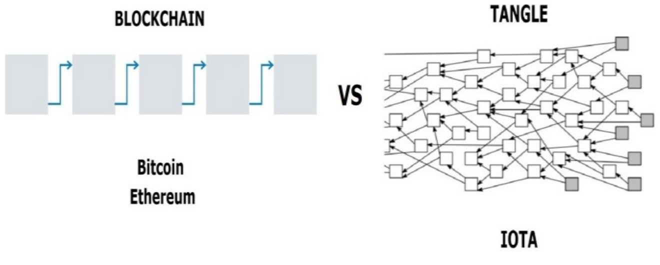


Fig. 2. The difference in BCT Ledger and IOTA Tangle ledger Structure

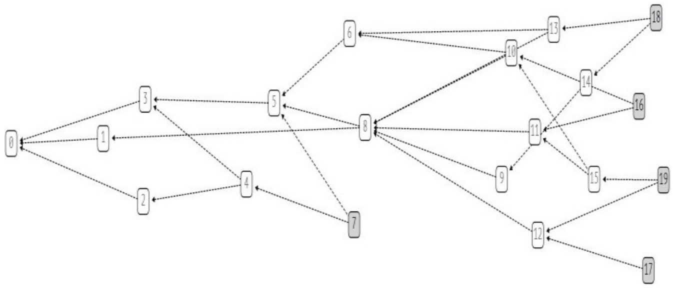


Fig. 3. Visualization of a Tangle

3) transactions :- contains IOTA tokens or data to be sent.

As shown in figure 3, there are different types of transaction in a tangle graph. At the vertices are the transaction and the edges represent the references between the transactions. Genesis transaction, which is approved by all the transactions. The Tip, highlighted in grey, which is the new transaction that is not yet approved by any other transactions. The confirmed transaction which is approved by other transaction either directly or indirectly.

As shown in figure 4, for a selected Transaction circled with red, Green transaction are the confirmed transaction approved directly or indirectly by the selected Transaction whereas yellow transactions are the transaction which directly or indirectly approves the selected transaction.

The idea behind the working of tangle is as follows:- for a transaction to be issued or added to the tangle graph, the user has to approve the previous two transactions. Thus, a user issuing a transaction is also contributing to the security of the IOTA network. If a transaction gets more number approval, then that transaction is added to the Tangle with higher amount of trust. While approving previous transactions, nodes check if the transactions are conflicting or not. If conflicting transactions are found then the transaction id not approved by the node and thus double-spending or conflicting transaction won't get added to the graph.

- Features of IOTA:-

1) Highly Scalable:- As the number of transaction increase, there will be more transaction available for approving the previous transaction, thus transaction approving rate increasing.

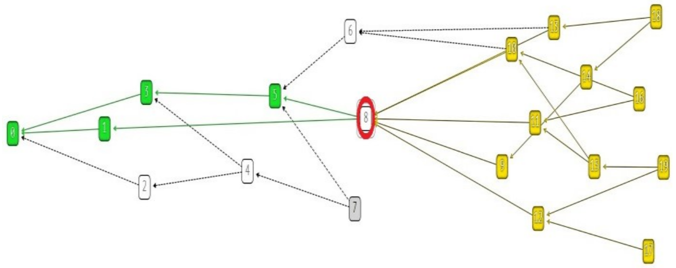


Fig. 4. Visualization of a Tangle with different types of transaction

2) Zero-fee transactions :- As there are no miners in IOTA, there exists no transaction fees, which enables us to perform micro-transaction, i.e. transaction of 1 cent is possible.
3) Quantum Immune :- For preventing attackers from stealing the users IOTA tokens, IOTA make use of a quantum robust signature scheme, named Winternitz one time signature scheme.
4) Secure data transfer :- Every data transferred using IOTA is encoded due to which the data storage , data transfer or reference are secured. All data is encoded, allowing secure data transfer, storage and reference
5) Low resource requirements :- As IOTA is designed mainly for IOT device which are resource constraint devices, thus it would require low resources.

A. MASKED AUTHENTICATED MESSAGING (MAM)

MAM, short for "Masked Authenticated Messaging", is one of the noteworthy feature and a potential step towards security of data in the Tangle developed by IOTA. It's name itself explains it working :-

- "Masked" : the message is encrypted;
- "Authenticated" : the message if verified and confirmed to be coming from authorized device;
- "Messaging" : the device publishes data in the form of a continuous stream of messages over the Tangle.

The Masked Authenticated Messaging (MAM) is an experimental module which is still undergoing peer review process. The MAM protocol belongs to the second layer of the Data Communication Protocol stack which allows additional functionality of transferring and accessing of an encrypted data stream over the Tangle. This encrypted stream of data may consist of messages delivered via zero value transactions over the Tangle. [12], [13], [16]

With the use, Masked Authenticated Messaging (MAM), the devices which are connected to the node of the IOTA Tangle can broadcast encrypted messages into a "Channel" from which anyone who has the required authentication keys can access and receive the message. [14], [15]

In IOTA, a user can send or publish a message over the Tangle, whenever the user has a data to send, only after doing some proof of work which allows the sent message to propagate the network. The sent message will be received in real-time once the message has reached the subscriber node and if the subscriber node is listening to that channel ID. [16] Every single message is a zero-value transaction sent to a unique address, where the address is chosen separately depending on the type of channel (public, private and restricted), and which can be used only once.

Every message in MAM channel is encrypted and signed with the Merkle tree signature scheme. Where the root of the Merkle tree is known as the Channel ID of the MAM channel. For each new message on the channel, a new

Merkle tree is generated. The messages have a reference to an address which contains the next subsequent message in the channel ,i.e. the root of the next Merkle tree, which allows the subscribers, who have access to a single message, can also find the subsequent future messages in the channel and easily follow the data stream. Since, only the next trees are referenced and not the previous ones due to which it give a kind of forward security to a channel. Thus the message streamflow is in only one direction. [15] The leaves of the Merkle trees are the hashes of the combination of the seed, which is private key that is only known to the channel publisher, and the index number of each leaf of the generated tree which starts from zero for the first leaf of the first generated tree and then increases for every new leaf. While decrypting a MAM stream message, the message is checked for valid signature and after verifying that the signature belongs to one of the leaf of Merkle tree and then message is decrypted. If the verification of the signature results in failure then the entire message is considered invalid. Inspite of MAM been an important component of the IOTA working , very less information is revealed to the public about its technical aspects as it is still under development.

Masked Authenticated Messaging gives the publisher the control over the accessibility of the data in the channel with three different modes, which are : Public, Private, Restricted. Regardless of which mode is been used, the subscribers/ users are provided with the root of the Merkle tree, which works as a reference pointer, which helps them to find the message from the Tangle. Each of the MAM modes is discussed below:-

1) *Public mode :* In this mode, the root of the Merkle tree is used as the address of the transaction to which the message is sent / published. A user who randomly tries an address may come across this message and once the user knows the Merkle root then the user can decrypt the message. Refer Fig. 5, it can be observed that the address and the root value both are same. This mode can be used in an application where the data can be accessible or anyone can view the data (i.e. the data is public) but with the additional property of the Tangle which is tamperproof and integrity.

2) *Private mode :* In this mode, the hash of the root of the Merkle tree is used as the address of transaction to which the message is sent / published. This provides an encrypted stream of data as any random user won't be able to decrypt the message as the address of the transaction and the root of the Merkle tree are not the same and also the root cannot be derived from the hash (Refer Fig. 6). The encrypted streamcan be accessed and viewed only by the publisher or the person who knows the root of the Merkle tree. The Private mode can be used for private communication between devices.

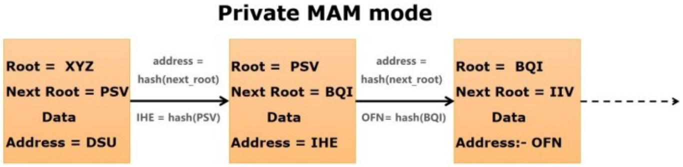


Fig. 6. Concept of Private MAM mode

3) *Restricted mode :* The restricted mode is similar to the private mode with an additional authorization factor (i.e. a key). In this mode, the hash of the root of the Merkle tree and a key (known as side key) is used as the address of transaction to which the message is sent / published. In this mode a publisher specifies the subscriber by sharing the key along-with the root, as the message is encrypted with the side key. Only if the correct side key if used then only the message can be decrypted (Refer Fig. 7). The message publisher can even change the sidekey by stopping the use of the current sidekey and without changing the channel as any subscribers can access the channel whenever needed. When a publisher changes the sidekey, then the new sidekey must be shared with subscribers so that they continue to view the data stream.

IV. System Implementation

The proposed work in this research paper aims to demonstrate the use of IOTA tangle in securing the important logs of a system. The idea of the implementation arises from the need of securing the logs of a system which are the utmost important evidence in forensics investigation of cyber crime. The logs are the first thing that a hacker modify or deletes, once he has conducted the crime inorder to remove any traces left. So for the forensic investigator to get the original and tamperproof log, the IOTA Tangle is been used. The logs of a systemis uploaded to the Tangle as soon as they are generated and later the investigator can even remotely access those file and monitor them without the doubt of the integrity of the files. As shown in figure 8, the consumer system logs are uploaded to the Tangle using the MAM Client API, every consumer has its own MAM Client API allowing them to upload the logs to the Tangle via different MAM channels. The appropriate root value and the sidekey must be shared with the forensic investigator so that the logs can be retrieved. While retrieving the logs on the Forensic Investigator (FI) side, the FI uses

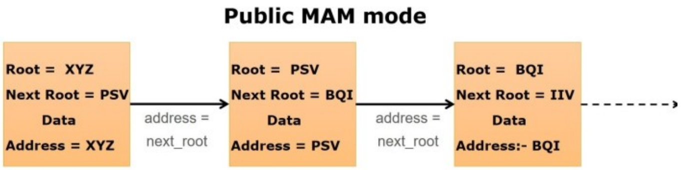


Fig. 5. Concept of Public MAM mode

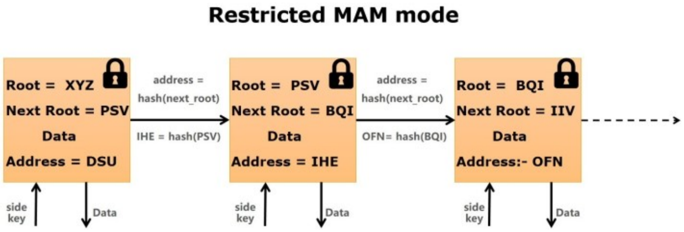


Fig. 7. Concept of Restricted MAM mode

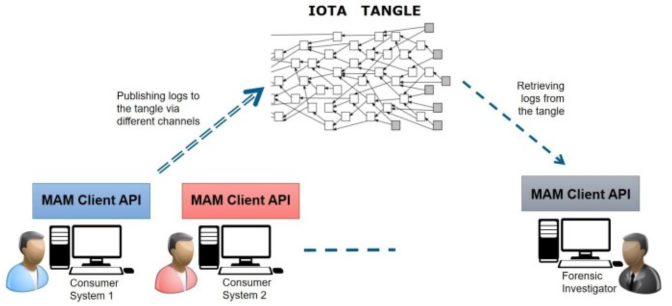


Fig. 8. System Architecture

appropriate root value and the sidekey depending on the MAM mode used, after which the FI can get the logs for monitoring without any doubt of the integrity of the files.

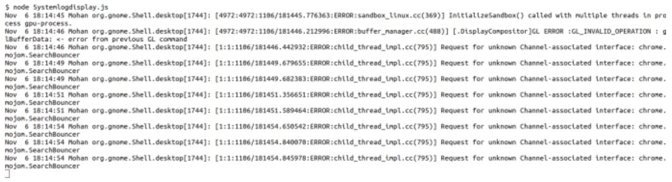
For developing our implementation the IOTA Reference Implementation (IRI) [11] is referred which is an open source Java software developed for implementing IOTA protocol, provided by the IOTA Foundation. The IRI software runs on node in the IOTA network, using which users or client can transfer data or IOTA token between each other. The IRI nodes have the following functions: validation of transaction, storing of validated transactions into the IOTA Tangle, Allows clients to connect to the IRI and get their Transactions attached to the Tangle.

The proposed work is being carried out on an open-source, cross-platform runtime environment Node.js used for the development of web applications. All Node.js applications are written in JavaScript and can be run on a wide variety of operating systems. The operating system used is Ubuntu 18.04 (Linux). Regarding the IOTA Reference Implementation data available on IOTA official blog page, the code is developed for the implementation in javascript.

In this implementation, IOTA is used for securing the logs of the system. Masked authenticated messaging(MAM) of IOTA is used which ensures only authenticated parties are sending the messages which are encrypted which ensure both confidentiality and integrity and got authentic tamperproof secure logs for forensic analysis. The MAM feature of IOTA is used in a restricted mode where only those parties having correct sidekey can access the data from IOTA Tangle
i.e only forensic investigator will be able to receive these logs, which preserves the confidentiality of logs. In this implementation, successfully implemented the securing of logs using IOTA where sending of logs of a system and retrieving the log data on a different system is shown.

A. Implementation Result

Step I :- Detecting system logs as soon as they are generated For this implementation, instead of viewing a complete log file only focused on the error message in the log file for ease of demonstration. Initially, only the error messages found in the log files of our system are detected and displayed. Here, as soon as there is any new error message in the log file, those messages will get displayed (Refer Fig. 9). This step is only



```
$ node Systemlogdisplay.js
Nov  6 18:14:45 Mohan org.gnome.Shell.desktop[1744]: [4972:4972:1106/181445.776363:ERROR:sandbox_linux.cc(369)] InitializeSandbox() called with multiple threads in pro
cess gpu-process.
Nov  6 18:14:46 Mohan org.gnome.Shell.desktop[1744]: [4972:4972:1106/181446.212996:ERROR:buffer_manager.cc(488)] [.DisplayCompositor]GL ERROR :GL_INVALID_OPERATION : g
lBufferData: <- error from previous GL command
Nov  6 18:14:46 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181446.442932:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:49 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181449.679655:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:49 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181449.682383:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:51 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181451.356651:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:51 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181451.589464:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181454.650542:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181454.840078:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181454.845978:ERROR:child_thread_impl.cc(795)] Request for unknown Channel-associated interface: chrome.
mojom.SearchBouncer
```

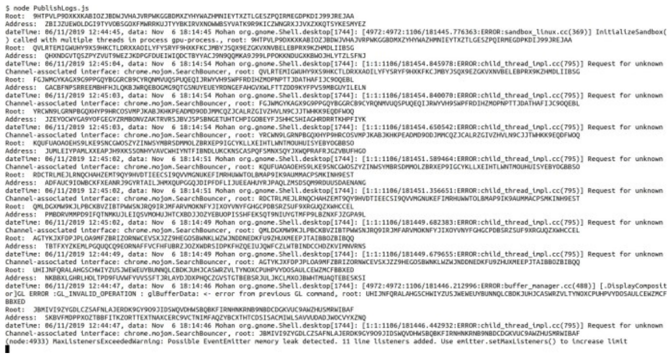
Fig. 9. Terminal output displaying the detected system log entries

for comparison between the logs got from the system and the log that are later retrieved.

Step II :- Publishing the logs to the Tangle After the detection of particular required logs, the logs are published to the Tangle using the IRI. The IRI gets a node setup and running to use. The messages of the log file are published as soon as they are detected by the system, which are properly time-stamped and published to the IOTA Tangle and mode and the sidekey both are hard coded in the implementation codes. Here, the restricted mode of MAM is used which uses the address as the hash of the Merkle tree root and the side key. Here the terminal displays the logs after they are published and attached to the Tangle. The Fig. 10 shows that every logs has a root, address, timestamp, which means that every log is treated as a new message in the MAM channel. Now anyone connected to the IOTA node can get these logs if he has the Root and the sidekey. Also, it was observed that the logs get published randomly but as they are correctly time-stamped, they can be retrieved in a correct sequence.

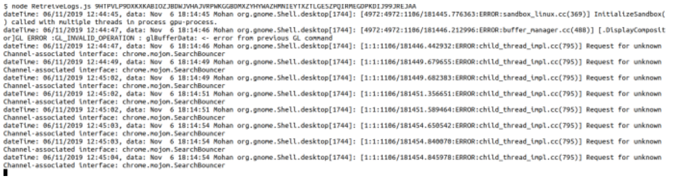
Step III:- Retrieving the logs from the Tangle

After the logs are published and attached to the Tangle, anyone with the Root and the sidekey can access these logs. A different system is used which is connected to the IOTA node, using IRI node.When the root is specified the logs are retrieved from that root and as every message has a reference to the next message so got the next logs in a sequence henceforth till the logs are published by the publisher node. As in Step II, mentioned regarding the random publishing of logs, but the logs are retrieved in a sequence in which they were obtained in Step I (Refer Fig. 11). Also, checked the logs in Step I and Step III were both identical. In this way a forensic investigator can remotely get



```
$ node PublishLogs.js
Root:  9HTPVLP9DXKXKABIOZJBDWJVHAJVRPWKGGBDMXZYHYWAZHMNIEYTXZTLGESZPQIRMEGDPKDIJ99JREJAA
Address:  ZBIJZUEWOLDGI9TYVOBSGOXFMWRRKUJTYYBKIRVXNOWMBSYVATK9R9KICZWNGRXJJVXZXKQTSYKESMYEZ
dateTime: 06/11/2019 12:44:45, data: Nov  6 18:14:45 Mohan org.gnome.Shell.desktop[1744]: [4972:4972:1106/181445.776363:ERROR:sandbox_linux.cc(369)] InitializeSandbox(
) called with multiple threads in process gpu-process., root: 9HTPVLP9DXKXKABIOZJBDWJVHAJVRPWKGGBDMXZYHYWAZHMNIEYTXZTLGESZPQIRMEGDPKDIJ99JREJAA
Root:  QVLRTEMIGWUHY9XS9HKCTLDRXXADILYFYSRYF9HXKFKCJMBYJSQX9EZGKVXNVBELEBPRX9KZHMDLIIBSG
Address:  QHXNDGVTQSZPYZVUT9WEZJKDPGFDUEIWIQDCTBYYACJ9N9QQMKA9J99LPPOKKNDUGXKBWOJHLYTZLSFNJ
dateTime: 06/11/2019 12:45:04, data: Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181454.845978:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: QVLRTEMIGWUHY9XS9HKCTLDRXXADILYFYSRYF9HXKFKCJMBYJSQX9EZGKVXNVBELEBPRX9KZHMDLIIBSG
Root:  FGJWMGYKAGX9G9PPGQYBGGRCB9CYRQNMVUQSPUQEQIJRWYVH9SWPFRDIHZMOPNPTTJDATHAFIJC9OQEBL
Address:  GACBFNPSRREEMBHFHJLQKBJWRQEBOGMG9QTGSNUYEUEYRDNGEFAHGVXWLFTTZDD9KYFPVS9MBGUYILELN
dateTime: 06/11/2019 12:45:03, data: Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181454.840078:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: FGJWMGYKAGX9G9PPGQYBGGRCB9CYRQNMVUQSPUQEQIJRWYVH9SWPFRDIHZMOPNPTTJDATHAFIJC9OQEBL
Root:  YRCWN9LGRNPBGQXHYP9HRCOSVMPJKABJKHKPEADMD9ODJMMCQZJCALRZGIVZHVLN9CJJTWHKK9EQDFWOQ
Address:  JZEYOCWYGA9YOFGEGYZRMBONVZAKTRVRSJBVJSPSBNGETUHTCHPIGOBEYFJSHHCSHIAGHRDRRTKHPFIYK
dateTime: 06/11/2019 12:45:03, data: Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181454.650542:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: YRCWN9LGRNPBGQXHYP9HRCOSVMPJKABJKHKPEADMD9ODJMMCQZJCALRZGIVZHVLN9CJJTWHKK9EQDFWOQ
Root:  KQUFUAOAOEHS9LKE9SNCGWOSZYZINWSYMBRSDMMOLZBRXEP9IGCYKLLXEIHTLWNTMOUHUISYEBYOGBBSO
Address:  JUMLEIYPAMLXXEAPJH9XKSSONHYVAVCWHIYNTFIBNDLUKCKNSCASPQFSMNXSQYJXWQPRAFRJGZVBUFHGO
dateTime: 06/11/2019 12:45:02, data: Nov  6 18:14:51 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181451.589464:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: KQUFUAOAOEHS9LKE9SNCGWOSZYZINWSYMBRSDMMOLZBRXEP9IGCYKLLXEIHTLWNTMOUHUISYEBYOGBBSO
Root:  RDCTRLMEJLRNQCHAHZEMT9QY9HVDTIEECSI9QVVMGNUKEFIMRHUWWTOLBMAP9IK9AUMMACPSMKINH9EST
Address:  ADFAUC9IOWBCKFXEANRJ9GYRTAILJHMXQUPGGQJDIPFDFLIJUEEAHUYRJPAQLZMSDSQM9RDUUSDAENANG
dateTime: 06/11/2019 12:45:02, data: Nov  6 18:14:51 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181451.356651:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: RDCTRLMEJLRNQCHAHZEMT9QY9HVDTIEECSI9QVVMGNUKEFIMRHUWWTOLBMAP9IK9AUMMACPSMKINH9EST
Root:  QMLDGXMW9KJLPBCKBVZIBTPWWSNJRQ9IRJMFARVMOKNFYJIXOYVNYFGHGCPDBSRZSUF9XRGUQZXWHCCEL
Address:  PMBDRVMMPD9IFQTNMKUJLEIQSVMOHUJHTCXBDJJOZYEBUOPISSHFEKSQT9NIUYGTMFP9LBZNXFJZGPA9L
dateTime: 06/11/2019 12:45:02, data: Nov  6 18:14:49 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181449.682383:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: QMLDGXMW9KJLPBCKBVZIBTPWWSNJRQ9IRJMFARVMOKNFYJIXOYVNYFGHGCPDBSRZSUF9XRGUQZXWHCCEL
Root:  AGTYKJXFDPJPLOA9MFZBRIZORNWCEVSXJZZ9HEGOSBWNKLWZWJNDDNEDKFU9ZHUXMEEPJTAIBBOZBIBQQ
Address:  TBTFXYZKEMLPGQUQCQ9EORNAFFVCFHFUBRZJOZXWDRSIDPKFHZQEIUJQWFCZLWTBINDCCHDZKVIMNVRNS
dateTime: 06/11/2019 12:44:49, data: Nov  6 18:14:49 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181449.679655:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: AGTYKJXFDPJPLOA9MFZBRIZORNWCEVSXJZZ9HEGOSBWNKLWZWJNDDNEDKFU9ZHUXMEEPJTAIBBOZBIBQQ
Root:  UHIJNFQRALAHGSCHWIYZUSJWEWEUYBUNNQLCBDKJUHJCASWRZVLTYNOXCPUHPVYDOSAULCEWZMCFBBXED
Address:  NKBBXLGHRLHOLTPD9FUWWFYVVSSFTJRLAYDJDXPHQCZGVSTGTBEBSRJULJKCLMXOJBWHTMUAQTEBESKSI
dateTime: 06/11/2019 12:44:47, data: Nov  6 18:14:46 Mohan org.gnome.Shell.desktop[1744]: [4972:4972:1106/181446.212996:ERROR:buffer_manager.cc(488)] [.DisplayComposit
or]GL ERROR :GL_INVALID_OPERATION : glBufferData: <- error from previous GL command, root: UHIJNFQRALAHGSCHWIYZUSJWEWEUYBUNNQLCBDKJUHJCASWRZVLTYNOXCPUHPVYDOSAULCEWZMCF
BBXED
Root:  JBMIVI9ZYGDLCZSAFNLAJERDK9GY9O9JIDSWQVDHWSBQBKFIRNHNKRNB9NBDCDGKVUC9AWZHUSMRWIBAF
Address:  SKBVFMDPPXOZTBBFITKZORTTEXTNAXCERC9VCTNIMFAQZYBCXTHTCDSISACMIWLSAVVUDADJWOCVYXZNQ
dateTime: 06/11/2019 12:44:47, data: Nov  6 18:14:46 Mohan org.gnome.Shell.desktop[1744]: [1:1:1186/181446.442932:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer, root: JBMIVI9ZYGDLCZSAFNLAJERDK9GY9O9JIDSWQVDHWSBQBKFIRNHNKRNB9NBDCDGKVUC9AWZHUSMRWIBAF
(node:4933) MaxListenersExceededWarning: Possible EventEmitter memory leak detected. 11 line listeners added. Use emitter.setMaxListeners() to increase limit
```

Fig. 10. Terminal output while Publishing system logs to the Tangle



```
$ node RetreiveLogs.js 9HTPVLP9DXKXKABIOZJBDWJVHAJVRPWKGGBDMXZYHYWAZHMNIEYTXZTLGESZPQIRMEGDPKDIJ99JREJAA
dateTime: 06/11/2019 12:44:45, data: Nov  6 18:14:45 Mohan org.gnome.Shell.desktop[1744]: [4972:4972:1106/181445.776363:ERROR:sandbox_linux.cc(369)] InitializeSandbox(
) called with multiple threads in process gpu-process.
dateTime: 06/11/2019 12:44:47, data: Nov  6 18:14:46 Mohan org.gnome.Shell.desktop[1744]: [4972:4972:1106/181446.212996:ERROR:buffer_manager.cc(488)] [.DisplayComposit
or]GL ERROR :GL_INVALID_OPERATION : glBufferData: <- error from previous GL command
dateTime: 06/11/2019 12:44:47, data: Nov  6 18:14:46 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181446.442932:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:44:49, data: Nov  6 18:14:49 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181449.679655:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:45:02, data: Nov  6 18:14:49 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181449.682383:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:45:02, data: Nov  6 18:14:51 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181451.356651:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:45:02, data: Nov  6 18:14:51 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181451.589464:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:45:03, data: Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181454.650542:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:45:03, data: Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181454.840070:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
dateTime: 06/11/2019 12:45:04, data: Nov  6 18:14:54 Mohan org.gnome.Shell.desktop[1744]: [1:1:1106/181454.845978:ERROR:child_thread_impl.cc(795)] Request for unknown
Channel-associated interface: chrome.mojom.SearchBouncer
█
```

Fig. 11. Terminal output while Retrieving the system logs back from the Tangle

the important logs of a system, without doubting the integrity of these logs. The only thing to think here is about the careful sharing of keys and root between the two parties.

V. ACKNOWLEDGMENT

The opportunity to work in the field of distributed ledger technology and Blockchain concept in itself is a great achievement. I take this opportunity to express my sincere thanks and respect towards the faculty and supporting staffs of the Department of Electronics & Telecommunication Engineering, and the Management, S.P.I.T., Andheri(W), Mumbai for providing the required support during the timeline of the research work. Lastly, I would like to thank my parent, colleagues and friends for their valuable suggestions, support and encouragement.

VI. CONCLUSION

The popularity of Blockchain has increased in various domains but in a few domains, like IoT and another machine to machine communication, its limitations are noteworthy. Different available distributed ledger technology for such domain which solves the existing limitation with its additional features have to be looked for. IOTA is such a distributed ledger technology for the IoT domain and machine to machine communication. Here the idea for securing the logs of a systemis focussed, IOTA suits best for this use case due to its ability to transfer zero-value transactions. This paper focuses on the IOTA Technology and discussed the details of its working. Also in this paper, discussed and successfully demonstrated the use of IOTA Tangle for securing the logs of the systemso that the investigator can get original untampered logs and successfully achieved the objectives of the paper. Masked Authenticated Messaging (MAM) of IOTA is used which ensures only authenticated parties are sending the messages which are encrypted which ensure both confidentiality and integrity and got authentic tamper-proof secure logs for forensic analysis.

REFERENCES

[1] Popov.S.,"The Tangle", Version 1.4.3,April 30, 2018.

[2] H. Anwar, "The Ultimate Comparison of Different Types of Dis- tributed Ledgers: Blockchain vs Hashgraph vs Dag vs Holochain", *101 Blockchains*, 2019.

[3] Fran Casino, Thomas K. Dasaklis, Constantinos Patsakis, "A systematic literature review of blockchain-based applications:Current status, classification and open issues", *Telematics and Informatics*, Volume 36,2019.

[4] User, S. ,"Blockchain: Cyber Security Pros and Cons." [online] *Apriorit*, 2017.

[5] Anwar, H.,"Blockchain Security: Premium Protection For Enterprises" [online] *101 Blockchains*, 2019.

[6] S.Rane and A. Dixit, "BlockSLaaS: Blockchain Assisted Secure Logging-as-a-Service for Cloud Forensics",*Communications in Computer and Information Science Security and Privacy*, pp. 77-88, Apr 2019.

[7] Belchior, M. Correia, and A. Vasconcelos, "JusticeChain: Using Blockchain to Protect Justice Logs,"*Lecture Notes in Computer Science On the Move to Meaningful Internet Systems: OTM 2019 Conferences*, pp. 318-325, Oct. 2019.

[8] "What is IOTA?," The Next Generation of Distributed Ledger Technology", *iota.org*.

[9] A.Fox, "Tangle-ELM Labs Research — ELM", *ELM*.

[10] A. Gal, "The Tangle: an Illustrated Introduction", *Medium*, 2019.

[11] IOTA Foundation, "IOTA Reference Implementation (IRI)", *iotaledger/iri,GitHub*.

[12] A.B. mushi, "IOTA: MAM Eloquently Explained", *IOTA News*, 17-Jun-2019.

[13] RuuviLab, "IOTA Masked Authentication Messaging," *RuuviLab*.

[14] B.Aldave, "Deciphering Masked Authentication Message (MAM),"*IOTA News*, 23-May-2019.

[15] M.Ø. Lindvall, "How is authenthicity and confidentiality maintained for MAM channels on the IOTA Tangle",2018.

[16] P. Handy, "Introducing Masked Authenticated Messaging,",*Medium*, 09-Apr-2018.

[17] K. Kent, S. Chevalier, T. Grance, and H. Dang.," Guide to integrating forensic techniques into incident response", *NIST Special Publication*, pages 800-86, 2006.