USING BLOCKCHAIN TO ENSURE GENUINE MEDIA ASSETS

M. Barroco[1], P. MacAvock[2]

[1] Orbitalize, Switzerland, [2] EBU, Switzerland

ABSTRACT

In a world where it is frighteningly easy to manipulate media content, fighting fake news and validating the source of that content to prevent it from being altered is vital.

This paper introduces the key concepts of Blockchain and Smart Contracts and proposes a design to ensure the traceability of media assets in a potentially untrustworthy environment. In this particular use-case, the distributed ledger keeps track of each operation which happens to an asset from its capture through to publication, including editing. This information can be checked against the ledger, in order to validate intermediate steps required to produce the resulting asset. This design ensures integrity of media assets including reliable chronological dating, and the validation of sources while preserving anonymity. At each stage of the workflow, manual or automated verifications are recorded, which creates chains of trust without central authority.

INTRODUCTION

It is all too familiar: a news report appears online with questionable origins; press reports of social media storms based on falsely attribute reports. It's becoming important to be able to distinguish genuine content and to identify the sources of that content. With modern editing tools, almost any individual is capable of editing video content at high quality, and immediately accessing an audience of billions through modern social networks – a potentially dangerous development for the media industry. When the infrastructure required to capture, edit and produce high quality content was the preserve of large media institutions, modification of that content was the preserve of jokes and novelty pranks. These trusted organisations also had exclusivity in being able to target large audiences rapidly. The term "viral" was an exclusively medical term 15 years ago.

Many examples of this phenomenon exist today. The BBC was forced to deny a story perhaps produced as a prank, but which went viral as outlined in the Telegraph, Horner (1). Manipulated images will become more and more difficult to detect as shown by Suwajanakorn et al. (2). They demonstrated how they used former US President Barack Obama's videos to synthetize his face and how they learned from audio how to fake lip-sync on a different video with the same audio.

News material has therefore to be traceable. It's not just the date and location of shooting, but a verified source and content accuracy are also now priorities for any media institution.

This paper will introduce the key concepts of Blockchain and propose a design of public blockchain to ensure the traceability of media assets from production to distribution in a potentially untrustworthy environment.

BACKGROUND

Blockchain

The concept of Blockchain emerged with the definition of bitcoin as specified in Nakamoto (3), with two key concepts:

- A distributed database to store all the transactions that have ever occurred in a blockchain network, which allows untrusted members of the network to send transactions securely without requiring a trusted central authority. See Figure 1.
- The concept of proof of work to secure entries in the blockchain. This avoids duplicated transactions and guarantees a single and unique state of the blockchain throughout the network as explained in Buterin (4). Therefore, multiple untrusted entities reach consensus on a single and shared history of transactions.

This history of transactions is called Blockchain or Ledger. Decisions and validation to update the history is based on consensus of the network and rules set at the genesis of the Blockchain.

Public and Private Blockchain

Blockchains can be either public or private as explained in Valenta and Sandner (7) :

- In a public Blockchain deployment, there is no trusted entity, and everybody is allowed to write and read the blockchain. Consensus is reached using mining (see Proof of Work below).
- For consortium and private Blockchains, the ability to append blocks is restricted to a limited set of participants and therefore rules set by trusted participants are applicable.

Ledger

The Ledger is composed of a list of ordered blocks. Each block gathers a set of transactions. Each block references the previous one as shown in Figure 2. This is why the technology is called Blockchain. The Ledger is stored by every node of the network.

Hash function

Each block is represented by a hash. It is generated using a cryptographic hash function, which is a mathematical algorithm which maps data of any size to a string of a limited number of bytes. Therefore, if you have the original data, it is simple to check if the hash is correct by hashing it again and

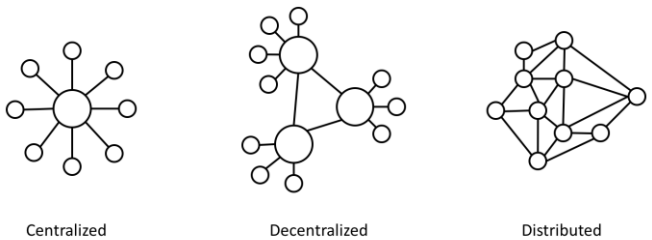

Figure 1 – Types of network

comparing the values. It is assumed to be practically impossible to find the original data from the hash itself except by trying all possible input combinations. In order to generate the block hash, the hash function takes as input the following information: The block id (position in the chain); the list of transactions; the hash of the previous block; additional headers; the Nonce value (more on this see below).

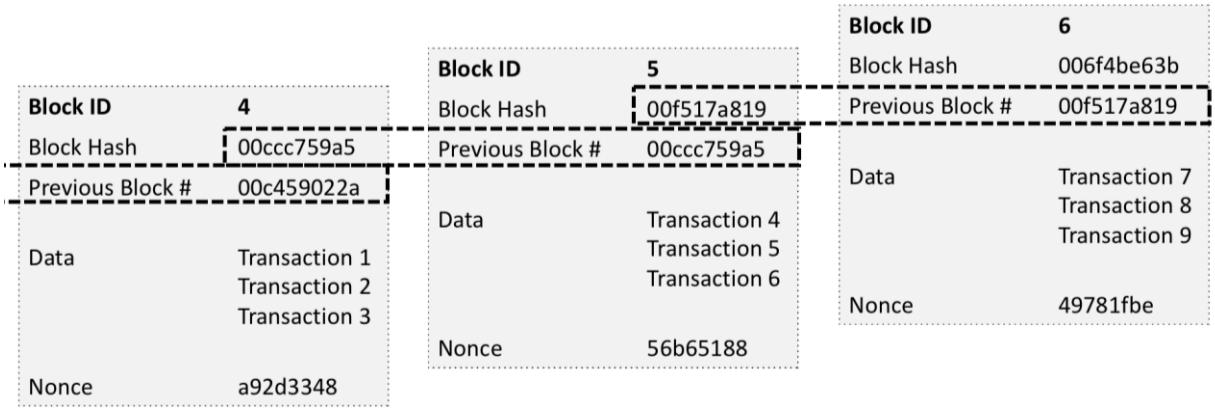

Figure 2 – Blockchain example

Proof-of-work

Since the network is composed of untrusted participants, it is important to slow down the ability to create new blocks and quickly generate a chain of blocks. This slowing down of the update rate of the blockchain allows the network to reach consensus and limits the ability to rewrite the history. This is achieved by using a proof-of-work system similar to Hashcash described in Back (5). The idea is to increase the complexity of finding a block by requiring participants wishing to store a block to find a hash with specific characteristics. As mentioned before, the only way to find a specific hash is to try all combinations until finding a compliant one.

Since a hash function will always return the same output for a specific input. Bitcoin adds a field called Nonce. The Nonce field is an arbitrary value which is appended to the block in order to generate the desired hash. The process of looking for a specifically complex hash by modifying the Nonce value is called *Mining*.

When a Nonce value is found so that the resulting block hash respects the complexity threshold, the block is broadcasted in the network in order to be added to the chain.

Immutable history of transactions

Changing the history by modifying a block would break the blockchain. Since the block hash of a specific block is based on the data of itself as well as on the hash of the previous one, a modification of a block would change its hash. Changing its hash would invalidate all the blocks coming after it in the blockchain. It will invalidate the next one because it will require generating a new hash for the next block and so on. Since computing blocks is computationally intense and thus expensive because of the proof-of-work, it's not feasible to generate blocks fast enough to create a branch longer than the accepted one on the network and thereby rewrite the history of transactions.

Consensus

The ledger is replicated across the network. If a conflict appears due to, say, two valid blocks being added at the same time and accepted by different parts of the network, the network will organically select the longest chain of blocks as being valid. Therefore, until the network reached a consensus (defined as a critical mass of nodes agreeing on the longest chain of valid blocks) a block can be invalidated.

Validation of Transactions

To be accepted by the network, the block needs to respect some rules. In the Bitcoin example, you can't create more output values than input values per transaction: "Reject if the sum of input values < sum of output values" (6). Each transaction can be composed of multiple inputs and outputs, so the system uses digital signatures to allow the owner of an input to sign the input value with the public key of the receiver in order to transfer the input to the next owner.

Scripts and Smart Contracts

The set of rules can be adapted to different use cases allowing the network to accept, reject or adapt according to the rules stored in the genesis block. These rules are coded using a programming language. For instance, Ethereum, as described in (4), generalizes the principles of Bitcoin to handle more use cases and encode arbitrary state transition functions. It allows, for instance, requiring multiple signatures to sign the inputs of a transaction.

Mining Incentive

In a public blockchain, a mechanism of reward is required in order to incentivize the mining process. Ether is an example of currency used to reward and to create incentives to miners and cover the cost of processing power.

Transaction Submission and Digital Signature

Based on asymmetric cryptography as explained in Stallings (8), every participant who submits a transaction owns a pair of keys, a private and a public one. Digital signature is used to preserve the integrity of an asset and to authenticate the author, who uses a secret private key to generate the signature and to claim the ownership. The public key is used to assess the integrity of the asset. If the asset was to change, the signature would no longer be valid.

NEWS USE-CASE

Context

Moving to the specific example of news assets, we would like to trace a piece of content from production through to distribution, keeping track of every transformation made to it. The transformation history should be immutable and every entry timestamped. Further, we can't assume any of the parties in the process are to be trusted. Finally, the process has to support the distribution of the media content at scale, with anyone joining the network to access the content. In particular, anyone can access the Blockchain and trace the transformation of any

content from its source, with members of the network free to stay anonymous or declare their identity.

Storage of media assets

Given the size of media asset files, it is not efficient to store them in the Blockchain. In order to keep track of changes to the files, a fingerprint of the content (for example using SHA-256 hash function) will be used to capture the state of a content at a given time. In the context of a public blockchain, the validation of the asset and its integrity would require having access to the media file. The InterPlanetary File System (IPFS) (9), which is a content addressable peer-to-peer distributed file system, could be a solution to store the assets at scale. Each file is split in blocks and gets a unique fingerprint called a cryptographic fingerprint. Immutable and permanent IPFS links can be stored into a blockchain transaction and would be available for public access, thus recording a timestamp and securing the content, without having to add the data on the chain itself.

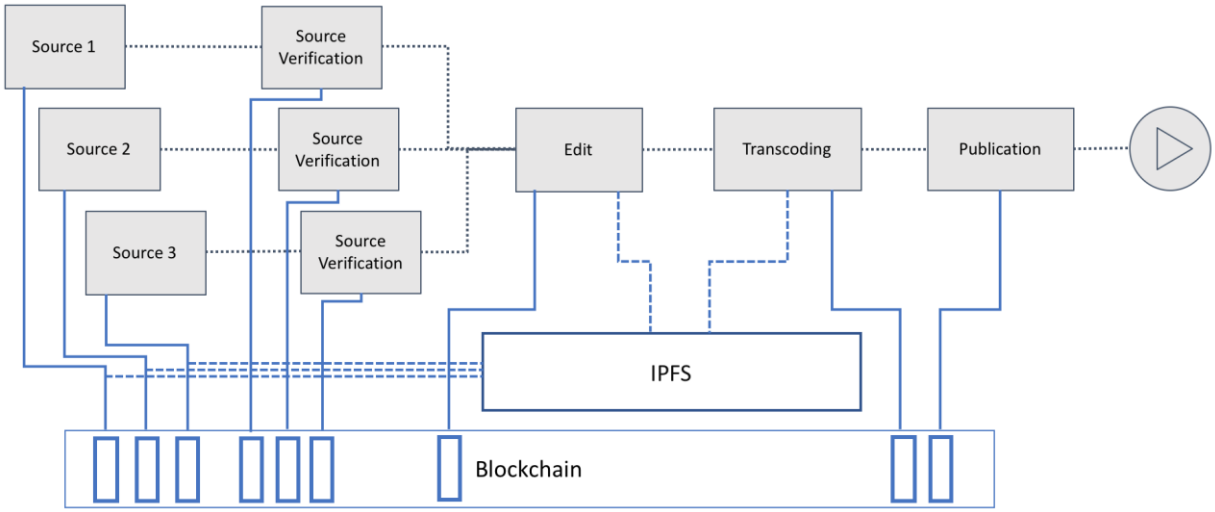

Figure 3 – Production to Distribution workflow steps stored in the blockchain

Transformation tracking

Given that we are tracking transformations and validation steps in an untrustworthy environment, (see Figure 3), we propose to use the Blockchain to record the different changes and actions on media assets using the operations as shown in Table 1. These can be encoded as a Smart Contract.

Table 2 shows an example of assets going through the workflow and the respective records in the blockchain.

Input [a]	Operation	Output	Description
[a] Every input is digitally signed and includes the Public Key of the owner of the operation.			
- Content fingerprint - IPFS location - Timestamp	***STORE***	Asset_hash	New assets can be registered to the blockchain by the owner of the Public Key. In order to validate the fingerprint, the content must be available to the blockchain network. The operation outputs a asset_hash which can be referenced in following operations.
- Asset_hash - Timestamp	***VERIFY***	Asset_hash (checked)	This operation records in the blockchain the fact that the owner of the Public Key has verified manually or automatically the content. (referenced by its hash)
- List(Asset_hash) - Transformed content fingerprint - IPFS location of the transformed content - Timestamp	***TRANSFORM***	Asset_hash	Transformed content is recorded in the blockchain like new assets. In addition, a list of asset_hash is declared by the owner of the Public Key to identify the assets used/transformed at this stage. (A new verification step may be required to assess the veracity of this entry)
- Asset_hash - Transcoded content fingerprint - IPFS location of the transformed content - Timestamp	***TRANSCODE***	Asset_hash	A ***TRANSCODE*** operation is similar to the Transform one, except that it takes only one hash as input.
- Asset_hash - IPFS location of the transformed content - Publication URL - Timestamp	***PUBLISH***	Asset_hash	The ***PUBLISH*** operation requires the signature of the input to be provided by a declared identity (see ***DECLARE IDENTITY*** below) in order to validate the ownership of the publication URL.
- Associated DNS - Timestamp	***DECLARE IDENTITY***		By default, everybody on the network is anonymous (self-certified keys). ***DECLARE IDENTITY*** links a public key to a specific DNS. The information of this block needs to be checked against a DNS entry in order to verify the authenticity of the declaration and avoid impersonation as presented in Hoffman (10). Information provided shall be used when validating the transaction in a block. Changing this record requires the same operation. The last entry in the ledger is authentic.
Note: Ethereum doesn't allow for external calls to validate a block. However, solutions less generic exist, as presented in (7), to use other programming languages able to call external resources.			

Table 1 – Operations to be recorded on the blockchain.

Description	Asset	Participant / Key	Transaction input (in addition to timestamp, public key and signature of data)	Transaction Output
Video capture and content registration	cam.mp4	Reporter or Journalist	SHA-256 (cam.mp4) ***STORE***	79c2dc17
News verification		News Agency	79c2dc17 (cam.mp4) ***CHECKED***	f69e7ad2
Editing of content	finalcut.mp4	Broadcaster	f69e7ad2 (cam.mp4 – checked) 515ad874 (cam2.mp4 – not checked) SHA-256 (finalcut.mp4) ***TRANSFORM***	3343af87
Transcode broadcast format to web format	index.m3u8 first.ts second.ts third.ts	Third party transcoder	3343af87 (finalcut.mp4 – not checked) SHA-256 (index.m3u8, first.ts, second.ts, third.ts) ***TRANSCODE***	d0410fc7
Web Publication		Content Delivery Network	d0410fc7 (transcode output reference) ***PUBLISH***	9159fb3d

Table 2 – Example of an asset operations' history

Genuineness check

At any time, it is possible to access the whole history of operations. Media assets recorded in the Blockchain are available in IPFS and their integrity can be automatically checked against the fingerprint stored in the blockchain. Each record can be verified by checking the list operations, which forms a chain of trust. If users are anonymous or not trustable, everybody can watch the content referenced in the blockchain as well as assess their genuineness using reliable timestamping. Any process of human or automated verification can be recorded in the blockchain using the ***Verified*** operation. In order to optimize the access, it would be possible to build an index of fingerprints and a graph of the operations to efficiently generate the history of a specific content by looking up its fingerprint. Ujo, which is a decentralized database of music rights and rights owners, is an example of blockchain using this method to quickly look up entries of the blockchain as described in Rouviere (11).

Notes on the use of Public Blockchain

Ensuring the integrity of the content and keeping track of the operations doesn't require Blockchain if the participants are all known. Indeed, a chain of digital signature could be stored in the metadata of the file. However, there will be no guarantee on the timestamp and it would be easy to regenerate a new file with new signatures. Indeed, as soon as the information is stored in the Blockchain, it would not be possible to modify it.

The access to the network and the ability to mine and participate to the consensus may be limited to a consortium of partners. This would mean deploying a private blockchain. In this case, roles may be allocated to different entities like shown in the table 2 above. Central authorities would be responsible to delegate different roles to entities. It would require a new set of rules and whitelist mechanisms embedded in the blockchain, which will for instance allow either everybody or only an accredited journalist to store content; source-checkers to store content and mark it as verified, etc.

POSSIBLE ISSUES

Many security and privacy issues are occupying the blockchain community today, in particular the proof of work mechanism, which drives the consensus. In Conti et al. (12), the authors summarize some key vulnerabilities. If scale and distribution is not maintained, some malicious attackers could take the control of the chain and possibly alter the blockchain or destabilize the consensus mechanism. Therefore, our proposed system could only work at global scale in order to limit the ability of a small set of entities to outrival the rest of the network by controlling a majority of miners.

POSSIBLE IMPROVEMENT AND FUTURE WORK

Additional work would be required to propose a more detailed approach to fingerprinting and assess the possibility to apply this mechanism on a frame base, essence base (Interoperable Mastering Format) and live content. In terms of protocol, we could extend the transactions set to handle other steps like manual and automated Quality Control checks or dubbing from third party providers.

CONCLUSION

Ensuring authenticity of media assets doesn't necessarily require Blockchain except where chronological dating is key and participants are untrustworthy. News assets are an example of such an instance. The deployment of such solution would enable the validation of sources (media asset and time) while preserving anonymity, but would require large scale adoption or the creation of a consortium to implement the solution at scale.

ACKNOWLEDGMENT

The authors would like to thank EBU colleagues, Mathieu Habegger and Maximilien Cuony for their technical input as well as Patrick Wauthier and Rasha Hasbini for their contribution on the use case definition.

REFERENCES

1. Will Horner, 2018, BBC forced to deny reporting outbreak of nuclear war after fake news clip goes viral, Telegraph, April 19, 2018.
2. Supasorn Swuwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman, 2017, Synthesizing Obama: Learning Lip Sync from Audio, ACM Transactions on Graphics, July, 2017, Vol. 36, No. 4, Article 95.
3. Satoshi Nakamoto, 2008, Bitcoin: A peer-to-peer electronic cash system, https://bitcoin.org/bitcoin.pdf.
4. Vitalik Buterin, 2013, A Next generation smart contract and decentralized application platform, White Paper, Ethereum.org, Web.
5. Adam Back, 2002, Hashcash - A Denial of Service Counter-Measure, http://www.hashcash.org/papers/hashcash.pdf.
6. Bitcoin wiki, 2017, Protocol Rules, https://en.bitcoin.it/wiki/Protocol_rules.
7. Martin Valenta, Philipp Sandner, 2017, Comparison of Ethereum, Hyperledger Fabric and Corda, FSBC Working Paper, June 2017,

https://medium.com/@philippsandner/comparison-of-ethereum-hyperledger-fabric-and-corda-21c1bb9442f6.

8. William Stallings, 1999, Cryptography and Network Security: Principles and Practice, Prentice Hall.
9. Juan Benet, 2014, IPFS - Content Addressed, Versioned, P2P File System (DRAFT 3), https://arxiv.org/abs/1407.3561
10. P. Hoffman, 2012, RFC 6698: The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA, Internet Engineering Task Force.
11. Simon de la Rouviere, 2016, Building Ujo #1: From The Technical Underground To The Future, blog.ujomusic.com, April 2018.
12. M. Conti, S. K. E, C. Lal and S. Ruj, "A Survey on Security and Privacy Issues of Bitcoin," in IEEE Communications Surveys & Tutorials, doi 10.1109/COMST.2018.2842460.