Tracking News Stories Using Blockchain to Guarantee their Traceability and Information Analysis

Francisco Jurado*, Oscar Delgado, Álvaro Ortigosa

Department of Computer Engineering, Universidad Autónoma de Madrid, Madrid (Spain)

Received 15 nov 2019 | Accepted 24 Feb 2020 | Published 25 June 2020

Abstract

Nowadays, having a mechanism to guarantee the traceability of the information and to monitor the evolution of the news from its origin, and having elements to know the reputation and credibility of the media, analyze the news as well as its evolution and possible manipulation, etc. is becoming increasingly significant. Transparency in journalism is currently a key element in performing serious and rigorous journalism. End-users and fact-checking agencies need to be able to check and verify the information published in different media. This transparency principle enables the tracking of news stories and allows direct access to the source of essential content to contrast the information it contains and to know whether it has been manipulated. Additionally, the traceability of news constitutes another instrument in the fight against the lack of credibility, the manipulation of information, misinformation campaigns and the propagation of fake news. This article aims to show how to use Blockchain to facilitate the tracking and traceability of news so that it can provide support to the automatic indexing and extraction of relevant information from newspaper articles to facilitate the monitoring of the news story and allows users to verify the veracity of what they are reading.

Keywords

Blockchain, Smart Contract, Traceability, News Stories, Journalistic Transparency.

DOI: 10.9781/ijimai.2020.06.003

I. Introduction

NOWADAYS, the news media are subject to the social scrutiny because of the lack of credibility, the manipulative media, the misinformation campaigns, and the propagation of fake news [1].

In [2], Lazer *et al.* point out that the main difference between fake news and true news relies on lack editorial norms and processes that ensure the accuracy and credibility of the information. Thus, to arrange a way that allows guaranteeing these editorial processes (or at least part of them) can suppose a big step in the fight against the above-mentioned issues. Also, in their work, Lazer *et al.* mentioned as an effective intervention to empowering individuals to evaluate the news. This way, end-users and fact-checking agencies like factcheck.org, snopes.com, politifact.com, hoax-slayer.com, truthorfiction.com, urbanlegends.about.com, newtral.es, or maldita.es, need a way to check and verify the information published in different media.

Is in this scenario where the *journalistic transparency* [3]–[8] has emerged as an essential concept to face the aforementioned issues. Applying the principle of journalistic transparency allows the readers to corroborate the information by directly follow the editorial processes and easily going to the source of the news and checking their veracity. So much so that transparency in journalism is currently a key element in performing serious and rigorous journalism. This transparency principle allows direct access to the source of essential content to contrast the information it contains and even to know whether it has been manipulated. Giving the right tools to guarantee the journalism process and allowing the users to verify the information are key concepts because humans tend to spread false news more than truth [9].

Additionally, allowing to check the journalism processes followed in the news stories, to track the evolution of the news reports and the relevant data and information it contains as they change over time, and therefore to trace how the related news evolves, constitute other instruments to face the previous issues. This is not only useful for end-readers but for fact-checking agencies and those tools that perform automatic indexing and extraction of relevant information of news. Once the information has been verified and fact-checked, these tools need a way to guarantee that the extracted data have not changed.

Combining these key matters in just one approach will suppose a starting point to solve the issues we mentioned at the beginning of the section.

However, as far as we know, there is no mechanism that allows guaranteeing to check the transparency and traceability of the information by monitoring the evolution of the news from the event that causes them, its possible manipulation during the dissemination process, and estimating the reputation and credibility of the media.

In this scenario, this article aims to detail the use of blockchain to facilitate the principle of journalistic transparency by allowing the tracking and traceability of news stories. Blockchain will be used as a tool to guarantee the traceability process and information analysis. This way, we will be able to support the automatic indexing and extraction of relevant information from newspaper articles to facilitate the monitoring of the news story and allows end-users readers and fact-checking agencies to verify the truthfulness of what they are reading.

The rest of the article is structured as follows: in section II we introduce the background and basic concepts; section III details the desired requirements we established for the proposal; section IV defines

* Corresponding author.
E-mail addresses: francisco.jurado@uam.es

the key elements of the proposal; section V details the operations we can perform; VI focuses on the real cost of using the approach; and finally, section VII provides some conclusions and future works.

II. Background

In this section, we provide the basis of our proposal, that is, the current situation about the application of the journalism transparency, and introduce blockchain and digital signature to guarantee the data integrity and authorship requirements we need to achieve the traceability of news.

A. Journalism Transparency

Journalism transparency is an ethical principle emerged as a solution to improve accountability and journalists' credibility [3], [8]. Following this principle, quality journalism should reveal how information was obtained, and thus, the readers and fact-checking agencies can go to its origins and check it [8], [10], [11].

In this context, some initiatives have raised in order to create guidelines and standards. Thus, the Journalism Trust Initiative (https://jti-rsf.org) appeared with the aim to create an agreed set of trust and transparency standards.

Similarly, The Trust Project (https://thetrustproject.org/) is a consortium of news companies that appeared with the aim of developing transparency standards to easily assess the quality and credibility of journalism and other standards for fairness and accuracy, a journalist's background, and the work behind a news story. Among their initiatives, The Trust Project defined what they call the Trust Indicators, a list of standardized disclosures about the news organization's ethics.

In line with this idea, the Newsroom Transparency Tracker (https://www.newsroomtransparencytracker.com/) is a tool that helps to determine the trustworthy of a news agency, by displaying the kind of public information available on a wide range of journalistic policies and practices.

Also using transparency indicator, we can mention the Transparent Journalism Tool (TJ-Tool) developed by the Spanish online journal Público (https://publico.es). This tool generates what the journal calls a transparency map. In essence, for each news item, it displays the author of the news, the dates of creation and edition, the list of reference documents, the people mentioned in the information, etc.

For its part, opting for the use of a social approach for fact-checking, WikiTribune [12] uses a collaborative approximation to write evidence-based news articles, where journalism professionals and volunteers collaborate together.

All in all, the main journals and news agencies claim to adhere to ethical principles and/or follow good practices and trust standards. Despite that, as far as we know there is no mechanism to ensure their fulfilment to guarantee journalism transparency, and this is where our approach of using blockchain comes into play.

B. Blockchain as the Key to Guaranteeing the Data Integrity Within the Chain

Essentially, a blockchain is a simple concept: a distributed and secure data registry, which guarantees the integrity of the stored information.

Despite its enormous potential, the *blockchain* concept has a modest and recent origin. As defined today, it was first described as an auxiliary technology of Bitcoin in 2009 [13], where it is used as a secure mechanism to store economic transactions between participants. Its recent explosion in popularity is due to the possibility of storing any type of digital data, guaranteeing its integrity.

This automatically enables many new possible uses for the technology: certification of documentation as mortgages, securities or any other official document [14], [15]; assets or intelligent objects [16], [17], which can make decisions based on the information stored in the blockchain; distributed security market, deposit and custody services [18], which would resolve disputes between customers and merchants; voting systems [19], [20]; or improvements in the supply chain for all types of products [21]–[23].

The blockchain technology provides some desirable characteristics, namely: **immutability**, **accountability**, and **availability** and **universal access**. These characteristics automatically raise the level of security, transaction verifiability, operational transparency and privacy of the secured information:

- *Security*: The decentralized nature of blockchains could guarantee that data remains *available* even in the case of failure of a substantial number of nodes. Due to its intrinsic immutability, a blockchain also assures the *integrity* of the data once it is recorded in it.
- *Transaction verifiability*: In a blockchain, any participant can validate transactions by itself, without relying on a centralized judge. Usually, the roles of the nodes are also distributed, in such a way that the centralization of interests is discouraged.
- *Transparency*: Usually, all participants in a blockchain share the same data and operations, whose security is guaranteed by a distributed consensus algorithm. This provides an accurate and consistent database for all participants, although their permission of access to information can be changed in some blockchains configurations.

In summary, the autonomous verifiability of transactions without the possibility of tampering or the necessity of a third-trusted party is especially useful in scenarios where the parties involved have conflicting interests. In this case, none of the parties can be the owner of the data, to avoid the possibility of manipulation.

C. Digital Signature to Guarantee the Authorship

As stated before, a typical blockchain guarantees the integrity of the stored data, but not its authenticity. This property must be provided with external cryptographic primitives, like *digital signatures*. A digital signature is a set of data associated with an electronic document and whose basic functions are:

- Identify the signer unequivocally.
- Ensure the integrity of the signed document, that is, guarantee that the signed document is exactly the same as the original and has not been altered.
- Ensure the *non-repudiation* of the signed document. This is a usually forgotten property, but very important in the proposed scheme.

To achieve this, each participant must create a pair of public/private keys, typically by using an x509 certificate. This certificate must be issued by a trusted Certificate Authority (CA).

III. Desired Requirements for the Proposal

This section details the functional and non-functional requirements we set while designing our proposal to achieve our goal.

A. Functional Requirements

In this article, we are aimed to address the issues about the lack of credibility, the manipulative media, the misinformation campaigns, and the propagation of fake news discussed in the previous sections, by enhancing the journalism transparency using blockchain.

To do so, we propose an architecture that will allow tracking the news stories and guaranteeing their traceability and information analysis.

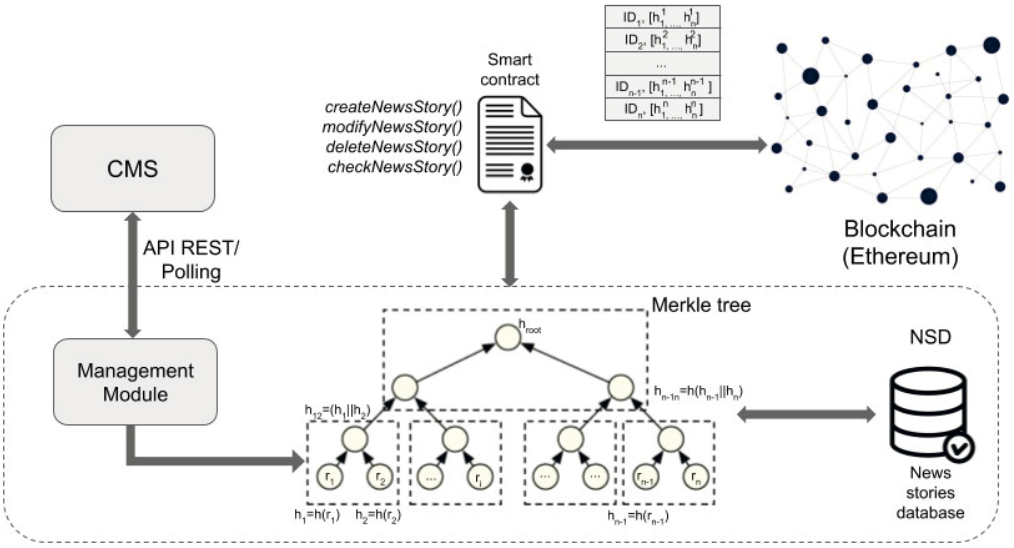


Fig. 1. Scheme of the proposal.

Thus, we established the next list of requirements for our approach:

- *Direct access to the source of information*: This requirement will easy the application of the journalistic transparency principle, what is essential to contrast the information contained in the news and to know whether it has been manipulated. This requirement is closely linked with the *transparency* characteristic of the blockchain.
- *Tracking changes of news stories*: This requirement will allow the traceability of the news, allowing analyze their evolution as they evolve over time. This requirement is related to the *transaction verifiability* in the blockchain.
- *Data and metadata integrity*: This requirement will be essential to guarantee the traceability process and information analysis. It is connected to the immutability characteristic in the blockchain.
- *Authorship insurement*: This requirement will provide a way to guarantee the authorship of the news, that is, the journalist or news agency who provides the information. This will be the key point where we will introduce the digital signature.
- *Date insurement*: This requirement will allow guaranteeing when the news was written and/or modified and, together with the previous requirement, by who. This is associated with the transaction verifiability in blockchain and the digital signature.

By achieving these requirements, we have ensured the journalism process, but also we have created a trust mechanism we have called the *fact-checking propagation*. That is, once a claim has been fact-checked, the linked data in the blockchain and the smart-contracts will guarantee and easily propagate the evidence gathered throughout the graph.

B. Non-functional Requirements

The main non-functional requirements of the proposal are:

- *Simplicity*: Keep the architecture as simple as possible, and with the lowest possible degree of coupling. This way the modifications to the existing systems are kept to a minimum.
- *Low computational requirements*: Keep the execution costs and processing time as low as possible, due to the huge potential number of data to be processed. To achieve this, our proposal will use an intermediate data structure called Merkle tree, in order to avoid the scalability problems of public blockchains.
- *Easy implementation*: Do not use complex smart contracts, which facilitates development and reduces execution costs. Smart contracts do not implement news treatment "logic"', but only the minimal functions needed to manage the storage of news stories (creation, modification, etc.)

This way, our proposal remains flexible and modular, by using a public blockchain as a secure storage mechanism to guarantee the integrity of the protected news stories.

IV. Architecture Key Elements

In this section, we describe the key elements of the proposed scheme to use public blockchains in the tracking of news. The rest of the subsections describes and analyzes the architecture and operation mode of our proposal, including its basic elements, namely:

- Management module
- Merkle trees
- News stories database
- Smart contract and blockchain

All these components are discussed in detail below, and a full scheme can be found in Fig. 1 in order to allow the reader a better understanding of the approach.

A. Management Module (MM)

This constitutes the central piece and the one in charge of managing all operations. It is also the entry point to the system, for which it will expose an API REST for integration with external tools. This API supports the basic CRUD operations (Creation, Read, Update, Delete) on each news story.

Thus, when the Content Management System (CMS) of news agencies or newspaper publishers creates a news story, the MM will support two modes of operation:

- *Active*: Active: the MM performs a periodically polling to the CMS to retrieve the new stories created during the last period.
- *Passive*: the CMS includes a small gateway module, which is activated when a new story is created and makes an appropriate call to the MM's REST API.

In any case, once the MM detects the creation of a news story, the certification process begins. This process is detailed in a later section.

B. Merkle Trees

Due to the transaction and storage costs and even to processing

capabilities, the protection of each individual news story with a public blockchain can be infeasible without an intermediate data structure.

For this reason, our architecture will make use of a data structure known as *Merkle tree* [24]. This construction is widely used in cryptography and computer science problems such as database integrity verification [25], peer-to-peer networks [26] and, of course, blockchains is not an exception [27].

A Merkle tree is a binary tree data structure in which every node contains the cryptographic hash of the concatenation of its child nodes contents. Due to this recursive way of constructing itself, the tree root contains statistical information of the rest of nodes, and the modification of any node content will cause the complete change of the value of the root. This way, the integrity of an arbitrary amount of data can be efficiently assured by arranging it in a Merkle tree form and securely storing the contents of its root node.

For every news story, our architecture maintains a Merkle tree storing the cryptographic hash of every necessary element in a node, namely: news author, publication/modification/deletion date, title, subtitle, body content, main picture, related documents (such as interviews, gazette publications, press conferences), etc.

To assure the root node of the Merkel tree we need what is called a *smart contract* (see section III.F). Therefore, when a new story is created, or an existing one is modified or deleted, the tree is re-calculated and the news root is updated in the blockchain.

As the hash function, the proposal can make use of any cryptographic hash function, such as the SHA3 family, which can produce outputs from 224 to 512 bits in length. However, in this work, we consider hashes of 256 bits per new story, which present a good balance between security and cost of storage.

C. News Stories Database (NSD)

As shown in the previous section, a Merkle tree only stores the hashes of the content, not the content itself. For this reason, the architecture includes the possibility of including an optional *News stories database* (NSD), which stores all the resources and elements of a news story. This way, the verification process, which will be described below, can be achieved more easily.

It is important to note that our proposal does not impose any restriction over this NSD. Typically, it could be a traditional relational database, internal or external to the organization.

D. Smart Contract and Blockchain

Finally, probably the most important element of the system is the *smart contract*, implemented in the Ethereum blockchain. This is the element that actually stores and secures the cryptographic proofs contained in the corresponding Merkle tree and implements the operations we can perform on them.

To demonstrate the viability and real performance of our architecture, the proposed smart contract has been implemented in Solidity language (see ANNEXE A. for the whole code) and deployed to the Ethereum Ropsten testnet at the address 0x8f737f448de451db9b1c046be7df3b48839673a1, where can be verified with any blockchain explorer like Etherscan.io. It is important to note that this is a basic contract, and that *should be used only for academic or educational purposes*.

The implemented operations in the smart contract are described in Table I. Of course, all these operations are authenticated, and can only be called by the owner of the architecture.

V. Operations of the Proposal

Once all the key elements have been analysed, this section describes the operation mode of our proposal.

A. Data to Work With

The core element is the concept of the *news story*. For each news story, journalists will provide its corresponding author, publication/modification/deletion date, title, subtitle, body content, main picture, and related content (such as interviews, gazette publications, press conferences) in any kind of format (text, audio, video), etc.

In our architecture, a news story (NS) is just an unordered set of resources **r** of length **n**, so that NS = (r1,..., rn). Of course, this set can be formed by any number and type of resources (typically text, images, videos, etc.).

From here, we set a protocol divided into two basic stages: **certification** and **verification**. The former describes the process that generates a cryptographic proof for each news story, which will allow us to verify its integrity at any time. The *verification* stage, on the other hand, allows any independent party to verify that the currently published news story matches exactly the initially published news and, if not, how, when and by whom it was changed. Both stages are described in detail below.

B. Certification

Essentially, the news stories are processed as follows (see Fig. 2):

1. A new story is created, processed and inserted in the CMS or similar software used by the journalist.
2. Our architecture periodically pulls this CMS and retrieves the news stories.
3. For each news story **NS**, a new Merkle tree $\mathbf{T}_{NS}$ is created, including

TABLE I. Operations Defined In The Smart Contract

Operation	Input arguments	Output	Description
createNewsStory()	NS_{ID}, h_{root}	-	This function is called to secure a news story with ID NS_{ID} and hash root h_{root} in the blockchain. If there exists already an entry with that identifier, it exits with no action.
modifyNewsStory()	NS_{ID}, h_{old}, h_{new}	*True* is modification is successful *False* if news story to modify does not exist	This function is called when a news story has been modified in some way since its inclusion in the system. It checks if the new story with ID NS_{ID} exists in the system and, if so, it changes its value to h_{new} and returns *True*. Otherwise, it returns *False*.
deleteNewsStory()	NS_{ID}	*True* is deletion is successful, *False* otherwise	This function deletes an existing news story with ID NSID. Due to the intrinsic immutable nature of public blockchains, data is not actually deleted, but marked as removed in the smart contract.
checkNewsStory()	NS_{ID}, h_{check}	*True* and timestamp of the certification if there exists a new story with ID NS_{ID} and hash h_{check}. *False* otherwise	This function is used during the verification phase, to check whether a news story was published as it appears today or has been modified.

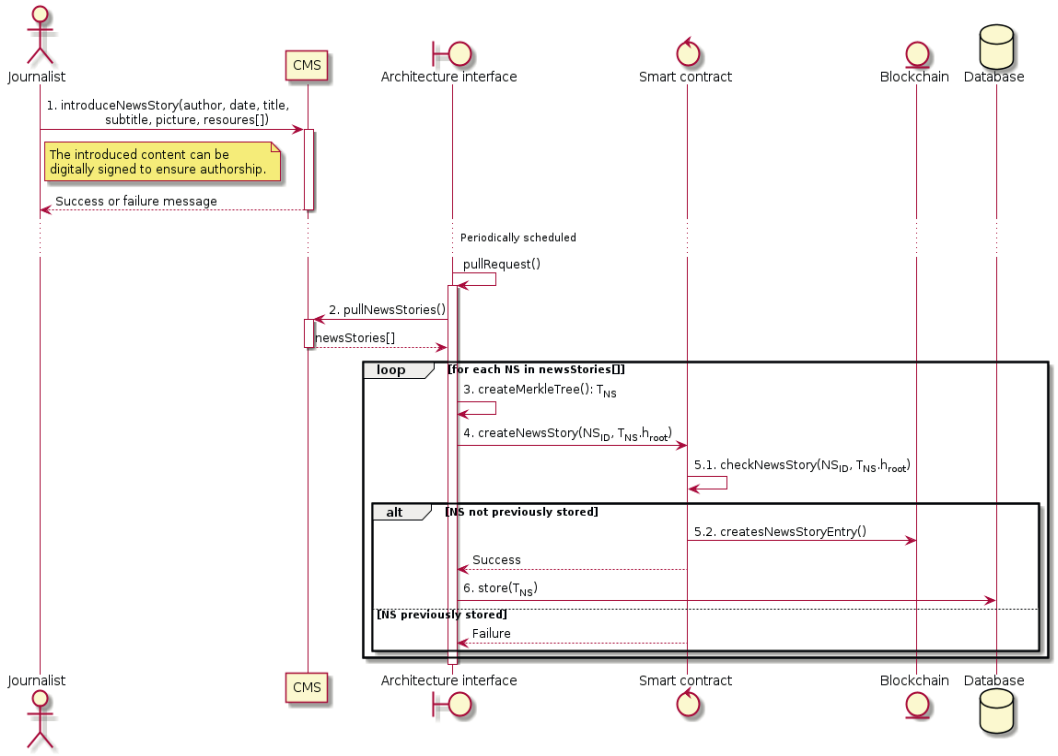


Fig. 2. Sequence diagram to perform the certification process while journalists introduce news stories in the CMS

each element ri of the NS as a leaf. The inclusion order items are arbitrary, but it has to be respected in the verification stage.

4. Once the construction of the $\mathbf{T}_{NS}$ is finished, its root leaf hash value, $\mathbf{h}_{root}$, is sent to the blockchain through the call to the *createNewsStory*(NS_{ID}, h_{root}) smart contract function.
5. The smart contract checks that no other NS has been previously stored in the blockchain with the same $\mathbf{h}_{root}$ and, if so, creates a new entry for this one.
6. Finally, the whole content of T_{NS} is optionally stored in a local or remote database (NSD), for further reference.

The previous process is repeated for every news story, so each news story is contained in a unique tree. Although the news story content will be typically kept in the journal's CMS, our architecture store the full contents in a local database for convenience.

Of course, in a typical news story lifecycle, this content is usually modified, corrected or improved. In these cases, it is necessary to keep an authenticated record of each change to ensure full traceability. Our architecture manages this situation in a very similar way to the certification process, but by calling the *modifyNewsStory()* function of the smart contract instead:

1. Same 1-3 steps than the certification algorithm.
2. Once the construction of $\mathbf{T}_{NS}$ is finished, its identification hash value $\mathbf{h}_{new}$, is sent to the blockchain by calling *modifyNewsStory*(*IDNS, hold, hnew*), where $\mathbf{ID}_{NS}$ is the ID of the new story to be modified, $\mathbf{h}_{old}$ is its previous hash root and $\mathbf{h}_{new}$ the hash of the modified version.
3. Then, this function checks that a new story with ID $\mathbf{NS}_{ID}$ and hash root $\mathbf{h}_{old}$ exists in its internal database, and is not marked as deleted. If so, it adds to the internal state of the new story the modified hash value, $\mathbf{h}_{new}$.

This way, our architecture keeps a full list of the different modifications a news story has suffered over time, knowing the exact time and date the modifications occurred, which provides a complete traceability capability.

This traceability is assured as follows. As can be observed in Fig. 1, the smart contract maintains an internal database for each news story NS. This database is implemented by using the mapping datatype of Solidity, which is essentially a *hash table*, i.e., a collection of tuples (**K**ey, **V**alue). Each entry has the form (NS_i, [h_i^i,...,h_n^i]), where $\mathbf{h}_i^j$ is the hash value corresponding to the *j*-th modification of the news story *i*-th of the system.

This way, the list [h_i^j] contains the history of changes that a specific news story has undergone since its introduction in the architecture. In addition, as each news item is digitally signed, it is also possible to guarantee its authenticity.

C. Verification

Any time later, a third party can verify the integrity of any news story previously published by performing the following steps (see Fig. 3):

1. Reconstruct the Merkle tree $\mathbf{T}_{NS}$ corresponding to the new story **NS,** using the same ordering for the resources that compose **NS**. As a result, a root value $\mathbf{h}_{root}$ is obtained.
2. Call the function *checkStory*(NS_{ID}, h_{root}) of the smart contract, passing $\mathbf{NS}_{ID}$ and $\mathbf{h}_{root}$ as arguments. This function checks if this value exactly matches any of the registers stored in its internal database and, if so, returns a *true* boolean value, along with a timestamp of the certification date of the information. Otherwise, it returns *false.*

As a result, we can obtain the immutability of the information stored in the blockchain (including the smart contract source code). If the result of the previous process is true, the verifier can be sure that the news story was not modified since its publication and certification.

VI. Cost of Using Blockchain for New Stories Tracking

As stated in previous sections, one of the main potential limitations for the integration of blockchain technologies is the cost of data storage

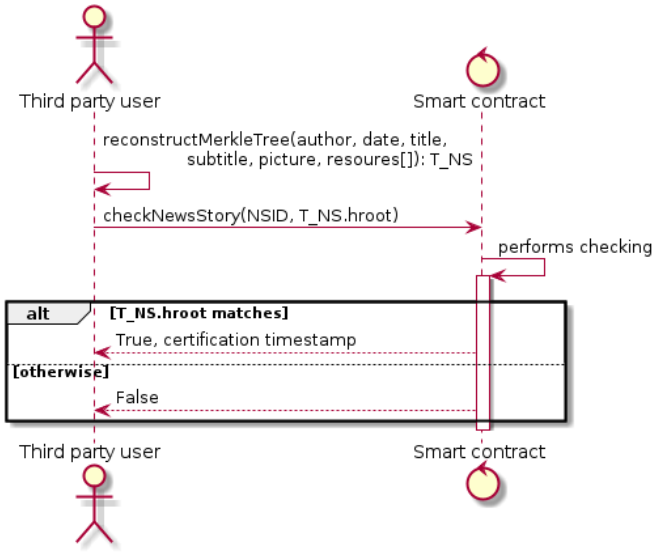


Fig. 3. Sequence diagram to perform the verification process by a third party user of agency.

and transactions processing. Therefore, to estimate the viability of our proposal, it is crucial to properly measure and minimize that cost.

The main reason is related to the execution of smart contracts in blockchains (like Ethereum). In order to reward the nodes that use their computing capacity to maintain the system, each instruction executed requires the payment of a fee in a cryptocurrency (called gas). Simple instructions (such as a sum) cost 1 gas, while others can cost significantly more (e.g., the calculation of a SHA3 hash costs 30 gas). On the other hand, the storage space is especially expensive (around 20k gas for every 256 bits).

There are essentially two approaches to store large volumes of data in public blockchains, which are presented below ordered in terms of complexity (from lower to higher), and economic cost (from higher to lower):

- *Full on-chain storage*: all data is stored, as-is, in the blockchain without any type of pre-processing. For example, news stories could be directly stored as a data structure in a smart contract.
- *Merkle trees*: data is stored off-chain, but it is preprocessed by constructing a Merkle tree structure, which reduces storage costs and increases the bandwidth. Typically, this is combined with data hashing, that is, storing only a hash of the data, in order to guarantee its immutability. The data itself can be stored off-chain in any other system: distributed (e.g., IPFS [15]), cloud, or local (as a common relational database).

In general terms, the storage space in public blockchains is especially expensive compared to computation, in order to discourage its abusive use. For this reason, the use of the first scheme, which is the most inefficient and costly, would commonly imply a prohibitive cost for most uses. As an example, Table II depicts the cost of reading and storing 1 Kilobyte of data in Ethereum in terms of gas units, ether, and US dollars.

TABLE II. Non-volatile Storage Costs In Ethereum. We Have Considered A Gas Price Of 1 Gwei (1 Gwei = 10^{-9} Eth), And 1 Eth = $140 (At The Time Of Writing, January 2020)

Operation	Gas/KB	ETH/KB	$/KB
READ	6,400	0.000032	$0.004
WRITE	640,000	0.0032	$0.448

TABLE III. Cost Calculated Considering 1 Mb Per New Story. Gas And Ether Prices Are The Same As Table II

Scheme	Operation	Cost (Per news story)	Execution time (average)
Both	Smart contract deployment	498274 gas ($0.06972)	19.20 secs
Direct storage	Certification	3.1·108 gas ($448)	10.55 sec
	Modification		
	Deletion	18850 gas ($0.0026)	11.66 sec
	Verification	-	-
Merkle trees	Certification	86848 gas ($0.014)	12.16 sec
	Modification		
	Deletion	18850 gas ($0.0026)	11.66 sec
	Verification	-	-

With these numbers, and considering a typical news story size of around 1MB (including texts and pictures), protecting just a thousand news stories would cost almost half a million dollars. Clearly, this is not a realistic option, discouraged not only in economic terms, but also for security and performance reasons.

For all these reasons, our proposal uses the second approach, based on Merkle trees. This is a much more elegant and efficient solution and, in fact, in most situations, the only viable option. Table III shows all the final results of our experiments.

As can be easily seen, the direct storage scheme is not a viable option in this case, since it would have a prohibitive cost in a real environment. On the other hand, our proposal shows affordable figures, both in economic and performance terms. This is due to the fact that the Merkle tree scheme needs to store only 256 bits per news story, regardless of its total size, which can be arbitrarily large and have any number of internal resources. This way, protecting the integrity of a news story costs around $0.014. This is clearly a very reasonable figure, even for large newspapers with a very large volume of news.

In any case, if necessary, the Merkle tree scheme could be adapted for processing any number of news stories at a fixed cost. The solution implies to create a hierarchy of trees, each containing a unique new story, and by grouping them in any number.

However, this scheme would be considerably more complex to implement and manage than our proposal. In this case, it would be necessary to create and store a (public) cryptographic proof with each new story, that would allow its later verification. This proof should be made available to the verifier.

Regarding the performance, the experiments show that this system is also viable. It is important to note that the tests have been carried out in a testnet, where the confirmation times are higher and have greater variability than in the mainnet. Times have been measured performing each operation ten times, discarding the minimum and maximum times, and calculating the average of the rest.

As can be seen, the execution time is slightly higher than 10 seconds for most of the operations, which seems an acceptable time for the usability of the system. Finally, the retrieval operation, necessary for the verification of a template, is a read-only operation and, therefore, free of cost. In addition, it can be also considered immediate in terms of execution time, due to the fact that the request is processed by the local Ethereum node, and it does not reach the network.

VII. Conclusion and Further Work

To determine a way to address the issues of fake news, disinformation campaigns, and the lack of credibility to which

journalists and media are exposed, throughout in this article we have presented a proposal that uses a blockchain to guarantee the principle of journalistic transparency that enables the tracking and tracing of news stories. This proposal arises as a real implementation alternative that guarantees the fulfilment of this principle, far from being a mere statement of intent by the media.

The proposal detailed in this article still has much room for improvement. For instance, the smart contract could be improved to support several content management systems simultaneously so that news from several agencies or editorial groups related to the same news story can be taken into account.

Another interesting improvement is the use of identities based on elliptic curves so that it is not necessary to use digital signatures dependent on external entities to ensure authorship, although the issue of digital identity from a legal point of view must be solved.

VIII. Annex A. Smart Contract Source Code

```
pragma solidity ^0.5.11;
pragma experimental ABIEncoderV2;

contract NewsAuth {

  struct NewsStory{
    uint256[] entries;
    bool exists;
  }

  /// Each template is indexed by an user ID
  mapping(uint => NewsStory) NS_database;


  /// Store a new template
  function createNewsStory(uint NS_ID,
                           uint256 h_root) public {
    // Check that there is no news story with this ID
    require(NS_database[NS_ID].exists != true,
            "News story already exists.");

    uint256[] memory _entries = new uint256[](1);
    _entries[0] = h_root;

    /// Add news story to internal database
    NS_database[NS_ID] = NewsStory({
          entries: _entries,
          exists: true
    });
  }

  /// Modify a user template
  function modifyNewsStory(uint NS_ID,
                   uint256 h_new) public returns (uint) {
    // Check that there is no news story with this ID
    require(NS_database[NS_ID].exists == true,
            "News story does not exist.");

    // Add modification to NS's history
    return NS_database[NS_ID].entries.push(h_new) -1;
  }

  /// Return an specific template
  function deleteNewsStory(uint NS_ID) public {
    delete NS_database[NS_ID];
  }

  /// Return a user template
  function checkNewsStory(uint NS_ID,
           uint256 h_check) view public returns (bool) {
    // Check that there is no news story with this ID
    require(NS_database[NS_ID].exists == true,
            "News story does not exist.");

    bool flag = false;
    for (uint i = 0;
         i < NS_database[NS_ID].entries.length;
         i++) {
      flag = (NS_database[NS_ID].entries[i]==h_check);
      if(flag == true)
        break;
      }
      return flag;
  }
}
```

Acknowledgment

This research work has been funded by the Madrid Regional Government through the project e-Madrid-CM (P2018/TCS-4307). The e-Madrid-CM project is also co-financed by the Structural Funds (FSE and FEDER).

References

[1] B. McNair, *Fake News: falsehood, fabrication and fantasy in journalism*. London: Routledge, 2017.

[2] D. M. J. Lazer et al., "The science of fake news," *Science*. vol. 359, no. 6380, pp. 1094-1096, 2018.

[3] A. Phillips, "Transparency and the new ethics of journalism," *Journal. Pract.*, vol. 4, no. 3, pp. 373–382, Aug. 2010.

[4] M. Karlsson, "The immediacy of online news, the visibility of journalistic processes and a restructuring of journalistic authority," *Journal. Theory, Pract. Crit.*, vol. 12, no. 3, pp. 279–295, Apr. 2011.

[5] M. Revers, "The twitterization of news making: transparency and journalistic professionalism," *J. Commun.*, vol. 64, no. 5, pp. 806–826, Oct. 2014.

[6] K. Chadha and M. Koliska, "Newsrooms and transparency in the digital age," *Journal. Pract.*, vol. 9, no. 2, pp. 215–229, Mar. 2015.

[7] T. P. Vos and S. Craft, "The discursive construction of journalistic transparency," *Journal. Stud.*, vol. 18, no. 12, pp. 1505–1522, Dec. 2017.

[8] M. Karlsson and C. Clerwall, "Transparency to the rescue?," *Journal. Stud.*, vol. 19, no. 13, pp. 1923–1933, Oct. 2018.

[9] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[10] D. Lasorsa, "Transparency and other journalistic norms on twitter," *Journal. Stud.*, vol. 13, no. 3, pp. 402–417, Jun. 2012.

[11] L. Morton, "Where are you coming from?," *Journal. Pract.*, vol. 9, no. 2, pp. 168–183, Mar. 2015.

[12] S. O'Riordan, G. Kiely, B. Emerson, and J. Feller, "Do you have a source for that?," in *Proceedings of the 15th International Symposium on Open Collaboration*, 2019, pp. 1–10.

[13] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system." 2009.

[14] A. L. Franzoni, C. Cardenas, and A. Almazan, "Using Blockchain to store teachers' certification in basic education in Mexico," in 2019 *IEEE 19th International Conference on Advanced Learning Technologies* (ICALT), 2019, pp. 217–218.

[15] J. C. Cheng, N. Y. Lee, C. Chi, and Y. H. Chen, "Blockchain and smart contract for digital certificate," in *Proceedings of 4th IEEE International Conference on Applied System Innovation* 2018, ICASI 2018, 2018.

[16] B. Notheisen, J. B. Cholewa, and A. P. Shanmugam, "Trading real-world assets on blockchain," *Bus. Inf. Syst. Eng.*, vol. 59, no. 6, pp. 425–440, Dec. 2017.

[17] Y. Yuan and F.-Y. Wang, "Towards blockchain-based intelligent transportation systems," in 2016 *IEEE 19th International Conference on Intelligent Transportation Systems* (ITSC), 2016, pp. 2663–2668.

[18] R. Khalil, A. Zamyatin, G. Felley, A. Gervais, and P. Moreno-sanchez, "Commit-chains : secure, scalable off-chain payments," *Cryptol. ePrint Arch.*, p. 642, 2018.

[19] K. M. Khan, J. Arshad, and M. M. Khan, "Secure digital voting system based on blockchain technology," *Int. J. Electron. Gov. Res.*, vol. 14, no. 1, pp. 53–62, Jan. 2018.

[20] R. Hanifatunnisa and B. Rahardjo, "Blockchain based e-voting recording system design," in 2017 *11th International Conference on Telecommunication Systems Services and Applications* (TSSA), 2017, pp. 1–6.

[21] D. Tse, B. Zhang, Y. Yang, C. Cheng, and H. Mu, "Blockchain application in food supply information security," in 2017 *IEEE International Conference on Industrial Engineering and Engineering Management* (IEEM), 2017, pp. 1357–1361.

[22] H. M. Kim and M. Laskowski, "Towards an Ontology-Driven Blockchain Design for Supply Chain Provenance," *SSRN Electron. J.*, 2016.

[23] H. Kaur, M. A. Alam, R. Jameel, A. K. Mourya, and V. Chang, "A proposed solution and future direction for blockchain-based heterogeneous medicare data in cloud environment," *J. Med. Syst.*, vol. 42, no. 8, p. 156, Aug. 2018.

[24] R. C. Merkle, "A digital signature based on a conventional encryption function," in Conf. on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology (CRYPTO), 1988, pp. 369–378.

[25] K. Mouratidis, D. Sacharidis, and H. Pang, "Partially materialized digest scheme: an efficient verification method for outsourced databases," *VLDB J.*, vol. 18, no. 1, pp. 363–381, Jan. 2009.

[26] H. B. Ribeiro and E. Anceaume, "DataCube: A P2P persistent data storage architecture based on hybrid redundancy schema," in 2010 *18th Euromicro Conference on Parallel, Distributed and Network-based Processing*, 2010, pp. 302–306.

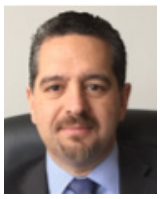
[27] C. Dannen, *Introducing Ethereum and Solidity: Foundations of cryptocurrency and blockchain programming for beginners. Berkeley*, CA, USA: Apress, 2017.

Francisdo Jurado

Francisco Jurado is Lecturer in the Computer Engineering Department of the Universidad Autónoma de Madrid, Spain. He received his Ph.D. degree with honors in Computer Science from the University of Castilla-La Mancha in 2010. His research areas include Natural Language Processing, and Computer Supported Collaborative Environments.

Oscar Delgado

Oscar Delgado received the B.S. degree in Computer Science from the Universidad Politecnica in 2002, and the PhD in Telecommunications Engineering by Universidad Carlos III de Madrid in 2011. His research interests include cryptology, network security, cryptocurrencies and blockchain. He leads the chair on blockchain technologies at Universidad Autonoma de Madrid funded by Grant Thornton.



Alvaro Ortigosa

Alvaro Ortigosa was born in San Carlos de Bariloche, Argentina, in 1968. He holds a Ph.D. on Computer Science from the Universidad Autónoma de Madrid in 2000, a M.S. on Computer Science from the Universidade Federal de Rio Grande do Sul in 1995 and a degree on Computer Science from the Universidad Nacional del Centro de la Prov. de Buenos Aires in 1993. He is director of the Research Institute for Forensics and Security Science of UAM since 2017 and Associate Professor at the Department of Computer Science of UAM since 2001. His main research lines are adaptive systems and user modeling, application of datamining for user model acquisition, personality and emotion detection through text and virtual social network analysis, and application of datamining to risk analysis. He has (co)authored more than 60 papers in international journals and conferences. Mr. Ortigosa is member of European Cybercrime Training and Education Group.