Article **Journal of Research and Development**
June 2019 Vol.5 No.15 26-37

Management of digital documents with encrypted signature, through the use of centralized PKI, and distributed using blockchain for a secure exchange

Gestión de documentos digitales con firma encriptada, mediante el uso de PKI centralizado, y distribuido utilizando blockchain para un intercambio seguro

ANTOLINO-HERNÁNDEZ, Anastacio†, FERREIRA-MEDINA, Heberto*, TORRES-MILLAREZ, Cristhian and OLIVARES-ROJAS, Juan Carlos

Tecnológico Nacional de México / Instituto Tecnológico de Morelia. Departamento de Sistemas y Computación
Instituto de Investigaciones en Ecosistemas y Sustentabilidad. UNAM campus Morelia

ID 1st Author: *Anastacio, Antolino-Hernández* / **ORC ID**: 0000-0001-6150-2934, **CVU CONACYT ID:** 21830

ID 1st Coauthor: *Heberto, Ferreira-Medina* / **ORC ID**: 0000-0003-0150-2355, **CVU CONACYT ID:** 67744

ID 2nd Coauthor: *Cristhian, Torres-Millarez* / **ORC ID**: 0000-0001-7619-0320, **CVU CONACYT ID:** 50277

ID 3er Coauthor: *Juan Carlos, Olivares-Rojas* / **ORC ID**: 0000-0001-5302-1786, **CVU CONACYT ID:** 394784

DOI: 10.35429/JRD.2019.15.5.26.37 Received: March 14, 2019; Accepted: May 12, 2019

Abstract

The project explores the use of digital documents as a response to the problems presented by physical documents, since they are at risk of partial or total loss. The solution is the digitalization that plays a very important role in society and the contemporary world. This helps sustainability and the preservation of natural resources. The security of the archives is a necessity that requires as solution to use the technology of public key infrastructure (PKI) to generate a digital document, besides registering the public and private keys of the personnel that has the legal power to sign them. These documents are stored on an official server and distributed among the registered hosts of the network. This certificate will help to detect changes in an unauthorized way, when comparing the document with the original. In this phase of distributed verification, the Blockchain technology will be used. Then the proposal is to build a tool to generate digital documents, in addition to managing public keys, transaction logs and records. The use of Blockchain will allow to establish and configure a Peer to Peer (P2P) network for a secure exchange.

Public Key, Blockchain, Digital document

Resumen

En este proyecto se explora el uso de documentos digitales como respuesta a los problemas que presentan los documentos físicos ya que éstos corren el riesgo de pérdida parcial o total. La solución es la digitalización que juega un papel muy importante en la sociedad y el mundo contemporáneo. Esto debido a que ayuda a la sustentabilidad y a la preservación de los recursos naturales. La seguridad de los archivos es una necesidad que requiere como solución utilizar la tecnología de infraestructura de llave pública (PKI, por sus siglas en inglés) para generar un documento digital, además de registrar las llaves públicas y privadas del personal que tiene el poder legal de firmarlos. Estos documentos se almacenan en un servidor oficial, y se distribuirán entre los hosts registrados de la red. Este certificado ayudará a detectar cambios de forma no autorizada, al compararse el documento con el original. En esta fase de verificación distribuida se utilizará la tecnología de Blockchain. Se propone entonces la construcción de una herramienta que permita generar documentos digitales, además de administrar las llaves públicas, bitácoras de transacciones y registros. El uso de Blockchain permitirá establecer y configurar una red Peer to Peer (P2P) para un intercambio seguro.

Llave pública, Blockchain, Documento digital

Citation: ANTOLINO-HERNÁNDEZ, Anastacio, FERREIRA-MEDINA, Heberto, TORRES-MILLAREZ, Cristhian and OLIVARES Carlos. Management of digital documents with encrypted signature, through the use of centralized PKI, and distributed using blockchain for a secure exchange. Journal of Research and Development. 2019, 5-15: 26-37.

* Correspondence to Author (hferreira@iies.unam,mx)
† Researcher contributing first Author.

Introduction

Currently, a large majority of people and companies, usually carry all the documentation in paper records (figure 1). However, this situation involves a lot of time for the attention, maintenance, search and presentation of said documents.

Figure 1 Physical records on paper, stored in a metal drawer
Source: Self made

To solve this problem, the documents collated in PDF files (Portable Document Format) will be kept (Adobe, 2016), through a system that will automatically manage these digitized documents, figure 2.

Figure 2 Example of a digitized document, to transform to PDF format
Source: Self made

This format is widely used on the Internet and was released as an open standard in 2008 by the creative company, Adobe Inc. This standard is published in ISO 32000 (2016). This standard is used, since a PDF file can contain text flow (encoded and / or compressed in several ways), images, fonts and various interactive elements, among others.

The management of these PDF files is done through their metadata, see figure 3, which can be stored in an information dictionary or viewed as a metadata flow, (Acrobat, 2016) (Adobe Metadatos, 2018).

Figure 3 Example of a PDF file, with data and metadata
Source: Self made

The problem to be solved is to identify the authenticity of a file in PDF format. To solve it, the system contemplates access to the metadata of the digitized files, for the validation of these, by means of the management of a digital signature (Digital Signature, 2018) (Digital Signature CR, 2018), which will serve to know if a file was altered by people outside the system.

The objectives of the implemented system are to guarantee the creation, modification and elimination of digital files, using digital signatures with the Hash encryption algorithms (Genveta Dev, 2018), see figure 4, SHA256 and RSA, respectively. Users and administrators must be provided with authenticity, reliability and integrity records.

The purpose of the mechanism proposed in this article is to take the appropriate measures to create and capture the digital files that meet the activity evidence requirements, see figure 5.

This is achieved by implementing watermarks per file for each user and system administrator.

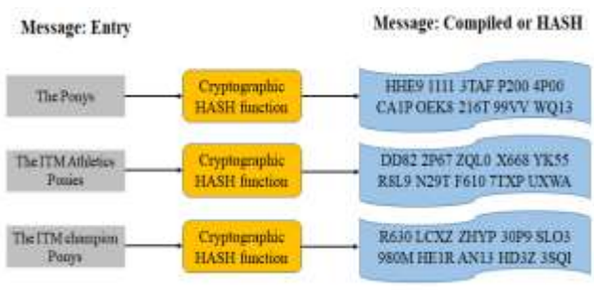

Figure 4 Example of operation of the Hash algorithm
Source: Self made

ISSN 2444-4987

Background

The concept of digital signature began its history in 1976, with the creators of computer cryptography Diffie-Hellman. Basically, the digital signature consists of a set of data associated with a message, which ensures the identity of the signer. In that year they presented their message authentication algorithm, which allows the protection of information. This algorithm uses symmetric keys for session encryption, so it is currently vulnerable. Subsequently, in 1978, the RSA algorithm emerged, which is the safest at present, since it uses the protocol known as asymmetric cryptography (Stallings W., 2005).

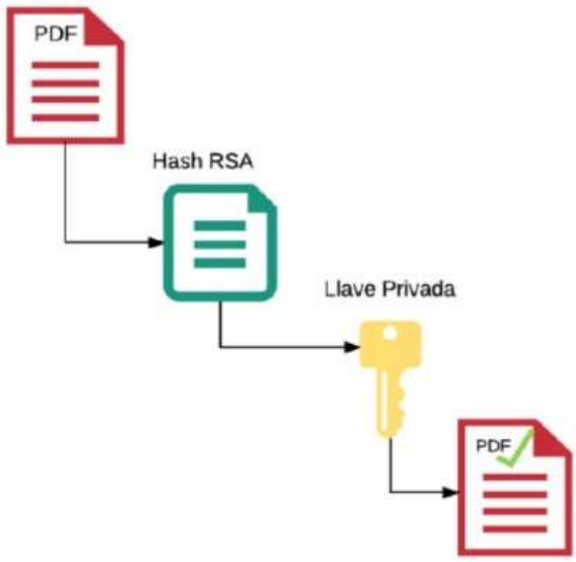

Figure 5 Structure of encrypted PDF files
Source: Self made

The RSA algorithm (from the authors: Rivest, Shamir, Adleman) uses two types of keys: public key and private key, unlike symmetric cryptography that only uses private keys. In the algorithms of symmetric encryption, we have the characteristic that the same key is used to encrypt and decrypt a message, the content of the message being vulnerable with the simple fact of sharing the same private key to protect the information.

Using asymmetric cryptography to protect a message sent between computers converts the same asymmetric cryptography into a digital signature. Instead of digitally signing the complete message, a summary or hash function of the message to be sent will be obtained, represented by a string of bits.

This hash will be encrypted and will serve to authenticate the identity of the sender and receiver of the message. When the message arrives at its destination, a different key will be used to decrypt the message, known as the public key. The summary of the document will be recalculated and as a consequence, if the values of the deciphered summary and the calculated summary are identical, the signature will be authentic and the message will be validated as complete.

Throughout time and technology, three types of digital signatures were developed, which are:

- Basic Signature: Includes a method of identifying the signer (authenticity). It her is the most vulnerable type.

- Advanced Signature: In addition to identifying the signer allows to guarantee the integrity of the document, preventing changes or alterations subsequent to the time of signing. PKI techniques are used.

- Recognized signature: It is the advanced signature executed with a DSCF (secure signature creation device) and covered by a recognized certificate (certificate that is granted after the face-to-face verification of the signer's identity).

Sometimes, this firm is called Qualified (Qualified) of the European Directive on Electronic Signature (UPV, 2017). The first legal antecedent of the digital signature was Directive 1999/93 / CE, of December 13, 1999, of the European Parliament and of the Council. There, a community framework for electronic signature was established.

To validate digital signatures are symmetric and asymmetric cryptography. As it was mentioned, nowadays, symmetric encryption is considered insecure, therefore, a digital signature has been implemented using an asymmetric encryption algorithm.

Cryptography

In computer science, it is the method of coding data, according to a specific algorithm and secret key so that only authorized users can re-establish their original form.

ISSN 2444-4987

It offers secure tools to ensure the authenticity, integrity and confidentiality of digital information (Herranz J., 2010).

Until 1976, the cryptography used was symmetric, when Whitfield Diffie and Martin Hellman introduced the concept of Public Key Cryptography and with the publication of the RSA algorithm in 1977 by Ron Rivest, Adi Shamir and Len Adleman, the most used cryptography was consolidated. Based on the public key, Herranz J. (2010).

Symmetric cryptography

Symmetric key encryption means that two or more users have a single secret key; this key will be the one that will encrypt and decrypt the information transmitted through the insecure channel.

This process is exemplified in Figure 6, where the secret key must have the two users (Emitter-Receiver), and with this key, the Issuer user will encrypt the information, send it through the insecure channel, and then the user Receiver will decipher that information with the SAME key that the issuer user has used.

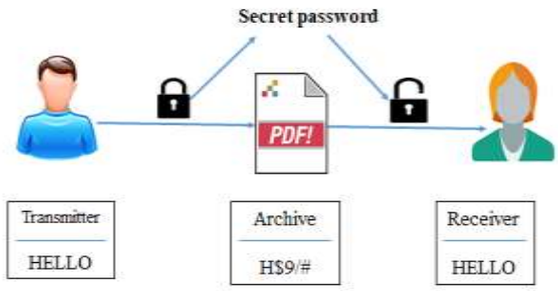

Figure 6 Symmetric encryption scheme
Source: Self made

With the asymmetric private key, the messages are encrypted and decrypted with the public key. In this way, the messages can be encrypted by the Issuer and transmitted to the Recipient, knowing the public key, that only he will be able to understand the message Ortega, J., López M. & García E. (2006). This procedure is described in figure 7.

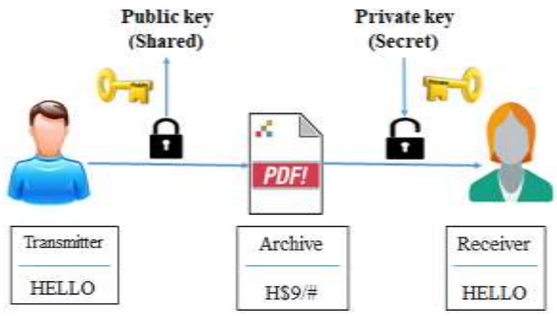

Figure 7 Scheme of asymmetric encryption
Source: Self made

The asymmetric encryption system has the purpose of signing PDF documents, certifying that the issuer is who he claims to be, this by signing with the private key and verifying, the receiving identity, with the public key.

Hash function

The hash functions are an encryption algorithm that, from an input either text, password or summary of a file, creates an alphanumeric output with a fixed 40-bit length. To regenerate the data in the chain, it is necessary to enter the same data again, which is why it is mostly used for passwords in databases. In figure 8, this procedure is exemplified.

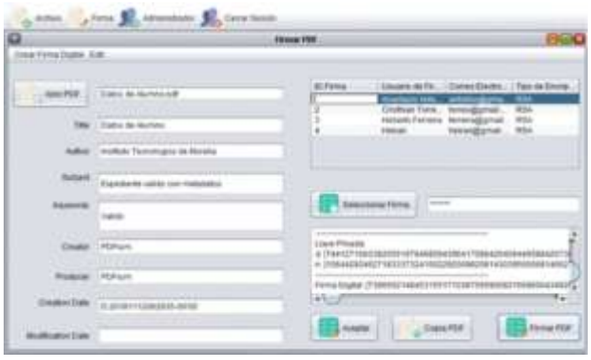

Figure 8 Application of the hash function to a digital document
Source: Self made

Asymmetric cryptography

Also known as public key cryptography, or public key encryption and private key. It is named like this, because in this cryptography each key consists of two keys: a public key, which can be shared or known by everyone; and another private, held by a single person, which is protected and saved by the user.

ISSN 2444-4987

Asymmetric algorithms are based on mathematical functions that are easy to solve in one direction, but very complicated to do in the opposite direction, unless the key is known. The public and private key are generated simultaneously and are linked to each other. This relationship must be very complex so that it is very difficult to obtain one from the other (De Luz, S., 2010).

Validation of documents using QR. The QR code (Quick Response code), emerged in 1994 as an evolution of the bar code, since it stores more information through a two-dimensional point matrix (crhoy.com, 2018), see figure 9, (TEC-IT, 2018).

Figure 9 Example of QR generated by the system
Source: (TEC-IT, 2018)

This special code is read by mobile devices or specific QR readers, which, when verifying the code, generate a text string, where the information stored is most commonly, a link to a website address or website or information that describes some particular topic.

In the project, the QR is used to obtain an address to a page in the Technological Institute system, with which a query is made and the document consulted is obtained according to an ID or folio. Which can be compared with the printed document or on file and compare its validity.

Figure 10 describes the methodology in a graphical way, the process in the generation of certified PDF documents. The process consists in scanning personal documents of the students, or generating a PDF document, such as a study kardex or a certificate of qualifications.

Once the PDF document is obtained, it is necessary to introduce control and information data to said file. The data are, for example, control number, creation date, system user ID, career, generation, etc. That they are practically internal control data, but if the document is intended to be delivered to the graduate or sent to another institution, it is certified digitally.

This digital certification consists of the following: a Hash code is added to the document, which is encrypted with the private key of the user responsible for issuing the official document.

After this digital signature, the QR code is added to the document for online validation.

The certified document is finally stored in the system database, see figure 10.

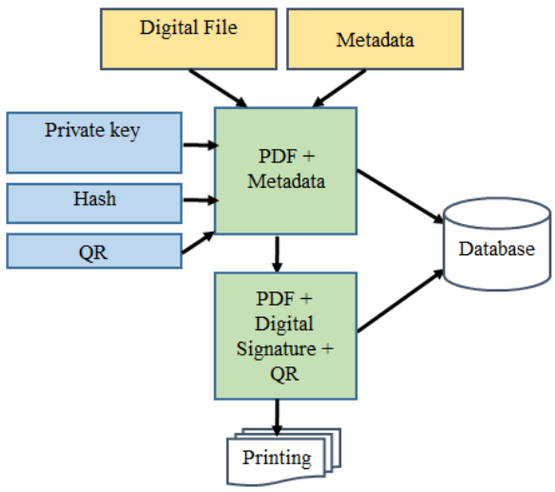

Figure 10 Methodology to generate secure signatures in documents
Source: Self made

Blockchain

We currently live in the fourth Industrial Revolution characterized by the use of ICTs in virtually all processes of human life.

Within ICTs, the most outstanding technologies are the following: artificial intelligence (AI), data analytics (AD), augmented reality (RA), Internet of Things (IoT), cloud computing (Cloud), printing in 3D and finally highlighting cyber security (CS).

ISSN 2444-4987

When talking about block chains, more than 90% of the time is associated with what BitCoin is: An encrypted electronic currency (cryptocurrency) from point to point and open source. There has been an increasing use of this technology to support cybersecurity.

Among its frequent uses include, in addition to the virtual currency, the handling of secure payments, authentication in IoT devices, smart contracts, electronic voting, validation of products as documents among many others. Some benefits are:

- Saving of time since transactions can be done in less time guaranteeing trust.
- Elimination of costs in the absence of intermediaries.
- Reduction of risks by avoiding cybercrimes such as manipulation and information fraud.
- Increased confidence in having a shared and traceable process.

Blockchain represents a historical database (DB) where you have all the information of the operations that have been carried out on a block of data from its origin to its current state. Queries of these operations can be made, although it is not fully optimized.

The DB grows rapidly over time as transactions are made. Figure 11 shows how a transaction is processed in a chain of blocks using the Hyperledger Fabric software that offers the facility to build nodes that exchange block chains consistently and safely (Androulaki E., 2018).

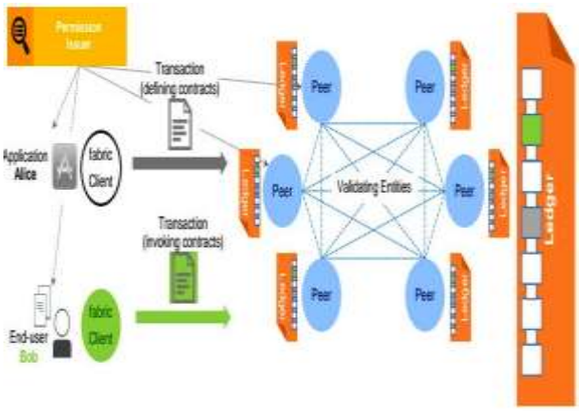

Figure 11 Hyperledger Fabric model
Source: (Androulaki E., 2018).

As shown in Figure 11, the Hyperledger factory is an authorized system in which all peer nodes are known (as opposed to the anonymous world of Bitcoin), block chains with a unique identity are managed. It is a system of permits where there are different roles for users and those that validate information (validators). Users invoke and implement their transactions (with the code that will be embedded in the block), which are then validated to create a new version of the block chain, that is, a single database (ledger in English). The key cryptographic element is an improved version of Byzantine fault tolerance practice (PBFT) known as a sieve.

Methodology

A PDF document is described through its metadata, and there may be multiple metadata flows from a single document. The structure of the files, when they are received with their respective metadata, go through the following process. The RSA encryption algorithm is applied, consequently it generates a private key that only the administrator of the digital signature will have. This key is used to encrypt the file and a public key will be used to decipher, thereby ensuring the integrity and authenticity of the document. Figure 12 shows a diagram of the mentioned procedure, to sign a document.

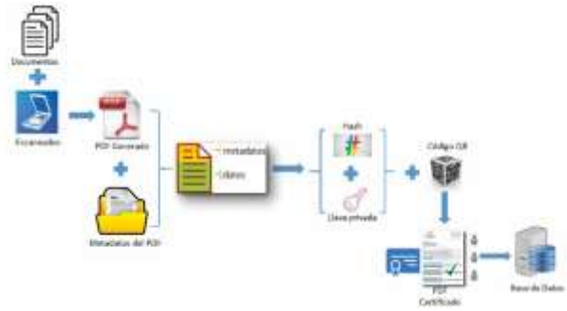

Figure 12 Scheme of the process of certifying documents
Source: Self made

To control the documents issued in PDF format, a program was implemented in the programming language Java SE (Standard Edition), using the free code library iText (2018), which was created to manipulate PDF, RTF, and HTML in Java. This software will read a document, get a summary of it, as well as the public and private key pair, which will be stored in a database within the institute.

ISSN 2444-4987

For the implementation of the digital signature, the Java SE programming language and the storage of the data in the MySQL manager were used. To be able to access the digital signature software, a login must be implemented, for which the data of the users that will be able to sign the documents digitally, as well as the resulting signatures, are stored.

For the storage of the users, the users table was designed, as shown in Table 1 and for the storage of the digital signatures, the table of signatures was designed, indicated in Table 2.

Key	Name	Type	Long Bytes	Description
PK	IdUsua	int	4	Id user for login
Nom	Name	varchar	50	User name
Usu	User	varchar	40	Profile
Clv	Clave	varchar	40	Password
TU	TpoUsua	int	4	Type of user, administrator, capturist, verifier
Act	Activo	int	4	Enabled to use application

Table 1 User information
Source: Self made

Key	Name	Type	Bytes	Description
PK	IdFirma	int	4	Digital signature ID
FK	UsuFinal	int	4	Id user who signed the document
NDo	NomDoc	varchar	50	Name of the document that was signed, metadata
HDo	HashDoc	varchar	128	Hash obtained encryption
PuK	ClvePub	varchar	50	Public key of the document
PbK	ClvePri	varchar	50	Document private key

Tabla 2 Información de las llaves digitales
Source: Self made

Implementation of blockchain

In general, blockchain mechanisms have been used mainly for the management of secure digital currencies (cryptocurrencies). In recent years, many applications have appeared, to ensure security in various areas of knowledge such as medicine, electronic voting systems, supply chain, among others.

That is why the idea of implementing a blockchain to guarantee security in transactions between servers that will exchange signed digital documents is born.

The problem to be addressed is to avoid malicious users, misconfigured devices, anomalies in documents, among many other external factors that affect transactions. Particularly, with this technology you can solve the problem of data manipulation (known as tampering).

On the other hand, within the project, a solution method is used, such as ElasticSearch, which is a distributed document-oriented search and analysis engine capable of solving a growing number of requests. Provides a full-text, distributed and multi-tenant search engine with a RESTful Web interface and JSON document standard.

This tool allows you to store, search and analyze large volumes of data quickly and almost in real time. It is used as the underlying engine / technology that drives applications that have complex search characteristics and requirements.

It is built on a search server based on Lucene (developed in Java) and using the Apache server, Angular framework. Some features offered by this search subsystem are (see figure 13):

- Access in real time: It allows us to access the documents that are being modified in real time.
- Scalability: Thanks to its design it allows us to scale horizontally and scale our servers (nodes) according to our needs.
- High availability: ElasticSearch clusters are able to detect which nodes are failing and reorganize to make data always accessible.
- Multi-Tenant: It allows us to operate on different indices at the same time and thus enhance our searches.
- It does not use schemes: It allows to work without a fixed structure of database.

ISSN 2444-4987

- Document oriented: Elasticsearch entities are stored as structured JSON files, where all the fields are indexed and we can include all the indexes in the same query.

- API: ElasticSearch provides RESTful APIs in JSON along with APIs for different languages.

- Text-based searches: ElasticSearch is based on Lucene, which increases text search capabilities, supporting gps and autocomplete.

- Conflict management: Prevents loss of data by simultaneously editing records.

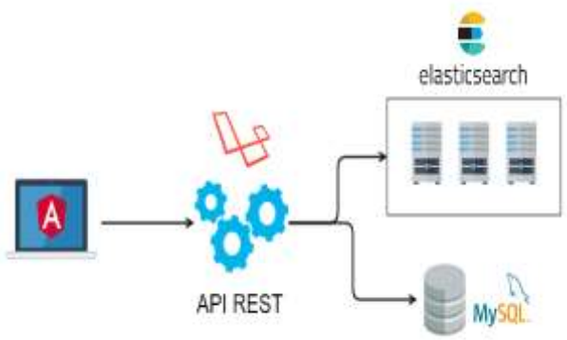

Figure 13 Tools used to search files with Slasticsearch
Source: (Espinoza-Avalos, Et. Al., 2017)

Returning to the use of block chains, it can be said that to implement the blockchain implementation, the following phases were used:

1. Network environment configuration. It is used to create the blockchain network.

2. Compilation software to build the chaincode. For the compilation system Hyperledger Fabric with Gradle is used (it is a compilation automation system that combines simple syntax to specify compilation components), along with the best features of Apache ANT and Apache Maven to create a powerful compilation system that is Easy to use.

3. Use of an http client: to invoke transactions in the chaincode. An http client software, which allows your chaincode to communicate with the REST interface of hyperchain's blockchain fabric. Your browser can issue an HTTP: GET, but to interact with the factory you need to be able to publish POST messages. This means that you need an HTTP client.

4. Start the network blocks.

5. Build the Java or C ++ client program that stores each transaction securely in a DB handler.

Results

The implementation of the Blockchain was done with three Hyperledger Fabric nodes, which in this case were three virtual machines using Docker, the general architecture of the developed solution is shown in figure 14.

Each docker container (virtual machine) refers to an organization. In our example, three educational organizations are being managed. Each organization has its client application developed in Java that is in charge of providing the scanned files in PDF, and the metadata as input.

In order for the organizations to communicate with each other, they are in charge of at least one entity that manages the communication with the other nodes of the blockchain network; in the Hyperledger architecture, this node is called peer. Therefore, the first step was to build the peers of the blockchain network; for this, it was necessary to configure the crypto-config.yaml file with the specifications of the topology of the network. Through this tool we can generate the certificates and keys for the organizations and the components within them (users and peers). In this case, for the three organizations, a single peer was created per organization and a single client per peer.

ISSN 2444-4987

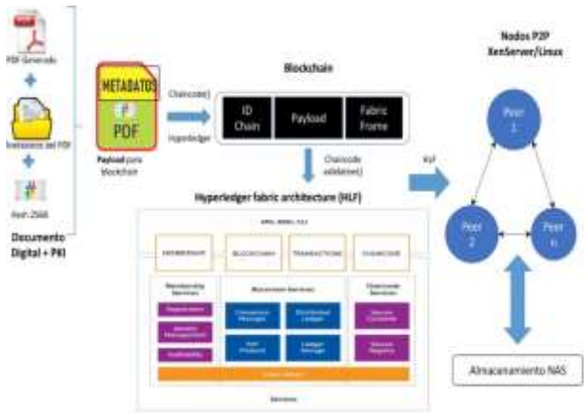

Figure 14 Blockchain model for digital file exchange, P2P
Source: Based on Bangbit.in (2018)

The second step was to define the coordinating node of the blockchain (it must be remembered that, although blockchain is a centralized mechanism, Hyperledger enters the category of private blockchain with permissions, which provides even more security). This coordinating node is defined in Hyperledger as the orderer, which is the main node in charge of the coordination of the organizations, see figure 13. The order was implemented within the organization of the Technological Institute of Morelia as another separate Docker container.

Then the channels and the MSP (Membership Service Providers) were configured. For this, it is necessary to configure the configtx.yam file as well as the consensus used by the orderer. In our case the kafka algorithm was used, which is tolerant to Byzantine failures.

Within this file in the Profiles section, communication channels are configured, in this case it was defined that all organizations can communicate with all of them, but this can be customizable.

For the creation of certificates you can use the fabricOps.sh script, which has an online command wizard that will guide us in the process for generating certificates. For the correct generation of certificates it is necessary to modify the function generateChannelArtifacts () with the appropriate routes of our project.

In our case we chose to use certificates and digital signatures already used in the PKI infrastructure, so we must modify the Hyperledger configuration files of each peer, to indicate the signatures and certificates to be used.

After generating the certificates it is necessary to modify the configuration of the Docker files that will generate the images of the peers, the certificates and the network ordering system. So that the blockchain network is consistent. This process has to be done in each new peer that connects to the network and in the order that will control it.

Then you have to create the channels, you have to make the peers join the channel, define the chaincode, instantiate it and be able to interact with it.

To execute the blockchain network, use the fabricOps.sh script with the start option. To create the channel, the peer channel create command is executed, indicating the files and variables of previously configured environments. For our case only one channel was created since all the peers of the organizations can communicate with each other.

The entry of each peer to the channel is done with the peer channel join -b command with the name of the created channel.

To install the intelligent contract you must execute the command peer chaincode install -n mycontract -v 1.0 -p route of the chaincode.

To instantiate the contract you must execute the command peer chaincode instatiate indicating in the variable Args, the initialization variables of our chaincode. At any time you can ask for some variable of our blockchain through the command peer chaincode query -C and indicating the contract and variables to consult.

To execute the intelligent contract, the peer chaincode invoke command must be used, indicating the contract route and its arguments (transactions).

ISSN 2444-4987

The implementation of the intelligent contract consists of two variables: series and folio. Three series A, B and C are configured, representing if they are grades, Kardex records and certificates. The other variable is a folio number that goes from 0 to 999. The chaincode validates the series and increases the number of folios.

When executing the chaincode and launching transactions the system behaves in a good way since there is no high load. The consensus is achieved quickly thanks to the kafka algorithm. The environment was programmed to make the consensus of nodes every minute, so you have to wait for this time to confirm transactions.

The size of the packages does not represent a big change, it has an average of 3.0 Mb per PDF with its metadata already included, so the ID size of the block and its Hashes do not consume more than 3.1 Mb.

Tests were conducted to measure the performance of the system with a period of 30 days under normal operating conditions in the middle of the semester (intermediate load period in school services) obtaining the following statistics from table 3:

Variable	Result
Number of Total Documents	2189
Total number of report cards	999
Total number of transcripts	757
Total number of records	433
Number of Modified Documents	288
Total number of blocks	1833
Number of transactions	2457
Average transactions per block	1.34
Final size of the chain blocks	66.58 Mb

Table 3 Variables counted during 30 days
Source: Self made

In a matter of safety tests the following were carried out:

1. It was validated that the smart contract will validate the folios and that they will not assign the same folios to different documents.
2. It was validated that, when the number of pages was finished, documents could no longer be stored in the ledger (Case of Qualification Records).
3. Confidentiality tests were conducted seeing that, without the proper keys and certificates, it is impossible to sign transactions in the blockchain without authorization.
4. Integrity tests were performed when trying to modify a metadata of an intermediate block, it could be noted that Hyperledger detects the change and reconstructs the chain of blocks with the replicas of other peers.
5. Finally, availability tests were performed with the Slowirs tool simulating Distributed Denial of Service (DDoS) attacks. For this, the firewalls of each peer and the orderer were temporarily disabled.

It was found that with at least one active peer the ledger can be replicated to the other dropped nodes. It was also observed that the ordering is vital and that if it falls, the transaction mechanisms are interrupted. Therefore, it is advisable to have more orders that help to balance loads, although for blockchain networks that are so small, it is not recommended for the operational load of infrastructure and performance for the blockchain network.

Conclusions

From the results obtained, when designing and implementing a system for the validation of digital documents in PDF format, it has been observed that this software can be used reliably to protect the integrity of the data, and thus, give more credibility and confidence to users and institutions.

Verification methods for the generation of digital signature using the RSA algorithm and QR code have been effective in terms of generation time and reliability that can be provided to a digital document.

The generation of the digital signature is relatively fast, the only drawback is the processing capacity of the peer node, since the larger the PDF document, the longer the digital signature is generated.

ISSN 2444-4987

The files generated with metadata have an average size of 3.0 Mb with key hashes of 256 bytes included, this information is used as blockchain payload, each block measures approximately 3.1 Mb, this is shared with the Hyperledger peers. It is observed the need to implement storage servers of large volumes of data to maintain the information of the blocks.

As a general conclusion, nowadays, it is important to have a system that can guarantee the integrity of the data and the application proposed in this article, it fulfills the established expectations, and it can even be used in the draft model of governance for implement the blockchain network Mexico (Michel G., 2018).

Finally, it is concluded that the use of paper in government institutions and companies can be significantly reduced, by using secure digital documents. Contributing to government projects oriented towards the ecology and sustainability of the environment.

Acknowledgments

To the National Technological Institute of Mexico / I.T. Morelia for her support to the project "Management of certificates of studies with digital signature through centralized PKI and using blockchain" with key no. 6758.18-P. To the Research Institute in Ecosystems and Sustainability of the UNAM, Campus Morelia, especially to the teachers Atzimba López M. and Alberto Valencia G., for their technical support. To the students of residences and social service for their help in the analyzes.

References

Acrobat (2016). Adobe Acrobat Inc. Recuperado el 25 de octubre de 2016. Acerca de Metadatos: http://help.adobe.com/es_ES/acrobat/using/WS58a04a822e3e50102bd615109794195ff-7c67.w.html.

Adobe (2016). Adobe Suite, recuperado el 12 de octubre de 2016 de: https://acrobat.adobe.com/mx/es/acrobat.html?promoid=KSBOO

Adobe Metadatos (2018). Adobe Acrobat Metadato, recuperado el 15 de enero de 2018 de: Propiedades y metadatos del archivo PDF. https://helpx.adobe.com/mx/acrobat/using/pdf-properties-metadata.html

Androulaki E. (2018). How Hyperledger Fabric Delivers Security to Enterprise Blockchain, recuperado el 1 noviembre de 2018 de: https://www.altoros.com/blog/how-hyperledger-fabric-delivers-security-to-enterprise-blockchain/

Bangbit.in (2018). Power of Hyperledger Fabric: Time to Make the Leap – An Enterprise Note. Recuperado el 1 de noviembre de 2018 de: https://bangbit.in/2018/04/13/power-of-hyperledger-fabric-time-to-make-the-leap-an-enterprise-note/

crhoy.com (2018). Código QR. Recuperado el 15 de agosto de 2018 de: https://www.crhoy.com/archivo/noticias-sobre/codigo-qr

De Luz, S. (2010). Criptografía: Algoritmos de cifrado de clave asimétrica. Recuperado de: https://www.redeszone.net/2010/11/16/criptografia-algoritmos-de-cifrado-de-clave-asimetrica/

Espinoza-Avalos F., Torres-Millarez C, Antolino-Hernández A, Ferreira-Medina H. (2017). Gestión de expedientes escolares mediante imágenes metadato para reducir el uso de papel y mobiliario. Tesis para obtener el grado de Ingeniero en Sistemas Computacionales. Tecnológico de Morelia. Asesor Cristhian Torres M. octubre 2017.

Firma Digital (2018). ¿Qué es la firma digital?, universidad Politécnica de Valencia, recuperado el 1 de agosto de 2018 de: https://www.upv.es/contenidos/CD/info/711250normalc.html

Firma Digital CR (2018). Gobierno de Costa Rica. Firma-Digital.CR, recuperado el 1 de septiembre de 2018 de: http://firma-digital.cr/que_es/

Genveta Dev. (2018). ¿Que son y para qué sirven los hash?. Recuperado el 17 de septiembre de 2018 de: https://www.genbeta.com/desarrollo/que-son-y-para-que-sirven-los-hash-funciones-de-resumen-y-firmas-digitales

Herranz J. (2010). Criptografía basada en atributos, IEEE España. U. Politécnica de Catalunya.

ISSN 2444-4987

ISO 32000 (2016). Estándar ISO 32000-1:2008, recuperado el 12 de octubre de 2016 de: http://www.iso.org/iso/catalogue_detail.htm?csnumber=51502

iText (2018). Toolkit iText PDF. Recuperado el 10 de agosto de 2018 de: https://itextpdf.com/

Michel G. (2018). Modelo de gobernanza para implementar la red de blockchain México. Estrategia Nacional Digital. Unidad de Gobierno Digital. Secretaría de la Función Pública.

Ortega, J., López M. & García E. (2006). Introducción a la criptografía: Historia y actualidad. Ed. Universidad de Castilla-La Mancha. España.

Stallings W. (2005). Cryptography and Network Security, Principles and Practices. Ed. Prentice Hall EEUU.

TEC-IT (2018). Barcode Tec It. Recuperado el 10 de agosto de 2018 de: https://barcode.tec-it.com/es

Torres-Millarez C., Antolino-Hernández A., Ferreira-Medina Heberto, Sarabia-Hernández J., Espinoza-Avalos F. (2017). Gestión de Expedientes Escolares Digitalizados, basados en Firmas Digitales para Verificar la Integridad, 1er. Congreso Estatal de Tecnologías Emergentes. Tecnológico Nacional de México/I. T. S. de Ciudad Hidalgo, Mich. Nov. 2017.

ISSN 2444-4987