Digital office

Cryptography

González Núñez Daniel Adrián
Hernández Castellanos César Uriel
Nicolás Sayago Abigail

Escuela Superior de Cómputo, IPN

April 08,2019

CHAPTER 1

Problem description

1.1 Problem description

The CEO of an important organization is trying to digitize several processes. In particular they are trying to generate the following documents automatically:

- ✓ Bill is an official written statement of the motions and resolutions taken in a meeting. It is brief but complete record of all discussions held among the members of the meeting.
- ✓ Memorandum is a note, document or other communication that helps the memory by recording events or observations on a topic such as may be used in a business office.
- ✓ Confidential memorandum is the same as a memorandum, but this must be kept in secret. This kind of document must be seen only by the sender and the receiver.

Any minute requires to be signed by every participating member in the meeting. Memorandums require the signature of the person who wrote the memorandum. Confidential memos, require the signature of the person who wrote the memorandum and also that only those persons authorized to see its content can read it. Design an application using cryptography to solve this problem.

1.2 Proposed solution

1.2.1 General idea

We would like to develop a web system that allows the users to generate the different types of documents listed in the requirements section (i.e. minutes, memorandums and private memorandums), to sign each document related to the meeting that they had attended previously and to be able to have a register of all the documents, and meetings, that the worker/employee has signed, or has been given permission to check, through their history as users of this system.

We strongly feel that a multi-level web application using specific cryptographic notions, like digital signature, will be part of the solution that we can use to develop a great system that fulfills all the requirements.

Modules

- ✓ **Signing in**
  Every person that will take part in a meeting and, specially, people that create documents must be registered in the system.
- ✓ **Create document**
  An user can create a document with a basic memo generating tool and can also sign it.(We assume that signs must be in the document)
- ✓ **Consult**
  An user can consult a document previously signed by themselves, or, send to them by an authorized user.

1.2.2 Technologies

We are thinking about working with:

- **Client side:** HTML, CSS, Javascript, JQuery.
- **Server side:** Java, MySQL

Like we previously mentioned, we are thinking about using the notion of "digital signature" to be able to give the opportunity to the users to sign each document with their given private key. But, to develop this idea further we think that we need to learn this subject in class first, before trying to integrate it to our application.

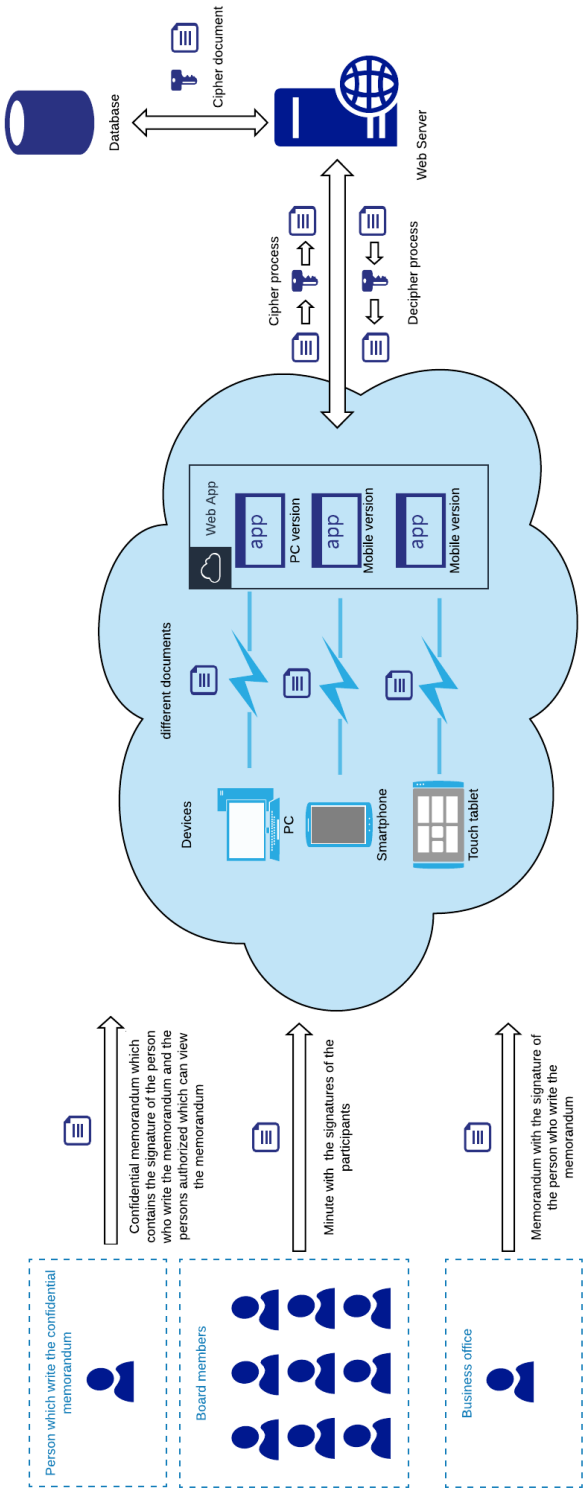
1.2.3 Questions about the project

- Should the documents be stored internally in order to consult them whenever we want?
- How many people are present in the meeting when the document is generated?
- Can the document be an automatically generated PDF?
- Which actors interfere with the system? i.e. Manager, assistant, worker, etc.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

1.3 Architecture

.



Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

The diagram above shows the complete process that our system works with.

On the far left side of the diagram we've got the users that create the different documents that our system will manage: The creators of both memorandums(normal and confidential) and all the members of the office that have been registered.

In the middle part we have the front face of the system. The connection between the users and the server happens when any user wants to do any operation, it could be a download of a memorandum, registering another user or signing a minute in which they participated prior to this.

Here our system works as a normal web application, the only thing that we have to ensure in order to fulfill the requirement is a secure connection between the client and the server, so the information that they upload corresponding to a confidential memorandum is always safe and only seen by the receivers.

Once the user had a successful connection to the server, and after completing an operation, the information is sent to the database. In the case of a confidential memorandum, the content of it is encrypted before being sent to the database to ensure that it can only be seen by the addressee.

More information about the encryption and decryption process is in the other diagrams.

If the user wants to see a memorandum that it was sent to them, they will have to upload their private key in order to check it. The other process work in the same way, depending on the operation the system will encrypt, or not, the information to be stored in the database.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

CHAPTER 2

Security services

2.1 Password storing

When it comes to handling personal data according to each member of the working team, everyone needs to have their own personal account to gain access to the system. Some information will be available for everyone to see: Full name, age, user, company role, attended meetings, etc. But, some other information, the one we use to authenticate each user, wont be public; one basic aspect that will have these characteristics will be each user's password.

In order to store user's passwords safely we cannot store them as common plain text in a password file; if an unauthorized user gains access to this file all the data in the system will be compromised, so we need a safe way to save this kind of information. The cryptographic primitive that we have planned to use to solve this problem is the use of a reliable **hash function**. We will use this hash function to generate a hash, let's call it $h(p)$, of our password p, that will be stored in the database instead of the common plain text. The use of a hash function assures us that if someone gains access to the data stored in the database, the attacker will be able to see the hashes corresponding to each saved password, but won't be able to know the original password as the hash function that will be implemented is a $one - way$ function.

To gain direct access to the system, the user will have to introduce their password(plain text), then the system will calculate the hash(using the same function used to store it previously) to compare the result with the password related to the user's information; only if the value matches then the user will be granted full access to their account.

Members of our team have worked with this kind of problem before in some other school projects but, had never faced a task like this because they didn't have to focus on security as much as other areas of the software developing process. We want to select a hash function so that we can use it thoroughly, and exploit every aspect of it to guarantee a high level of security to this aspect of our project.

2.2 Digital Signature

At each meeting the documents will be generated by one member of the working team. The requirements from the previous section set the following characteristics for each document:

- ✓ Minute: Each member of the working team has to sign the minute.
- ✓ Memorandum: It requires the signature of the person who wrote the memorandum.
- ✓ Confidential memorandum: Required the signature of the person who wrote the memorandum.Plus, it can only be seen by authorized workers.

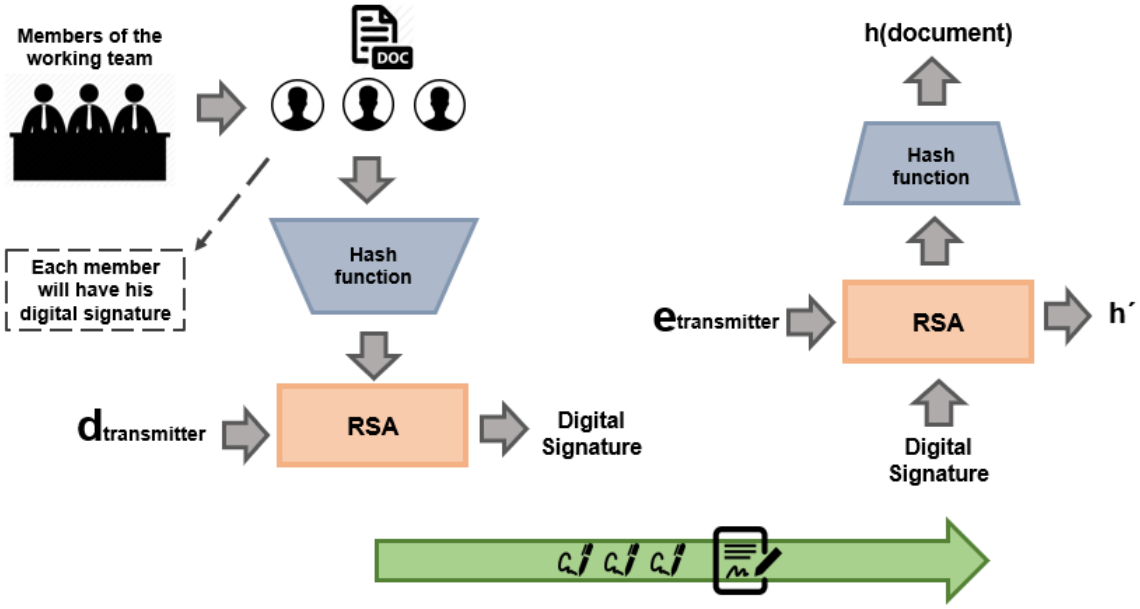
As we can see, it is an essential procedure to sign each document. The cryptographic primitive that we have planned to use to fulfill this requirement is the utilization of a reliable hash function. For that, we need a public key scheme so we selected the RSA public-key cryptosystem.

In the following sections we will explain the process that we have planned for each document. It's important to remember that the digital signature provides us of the following cryptographic services: **non-repudiation, integrity and authentication.**

Minute

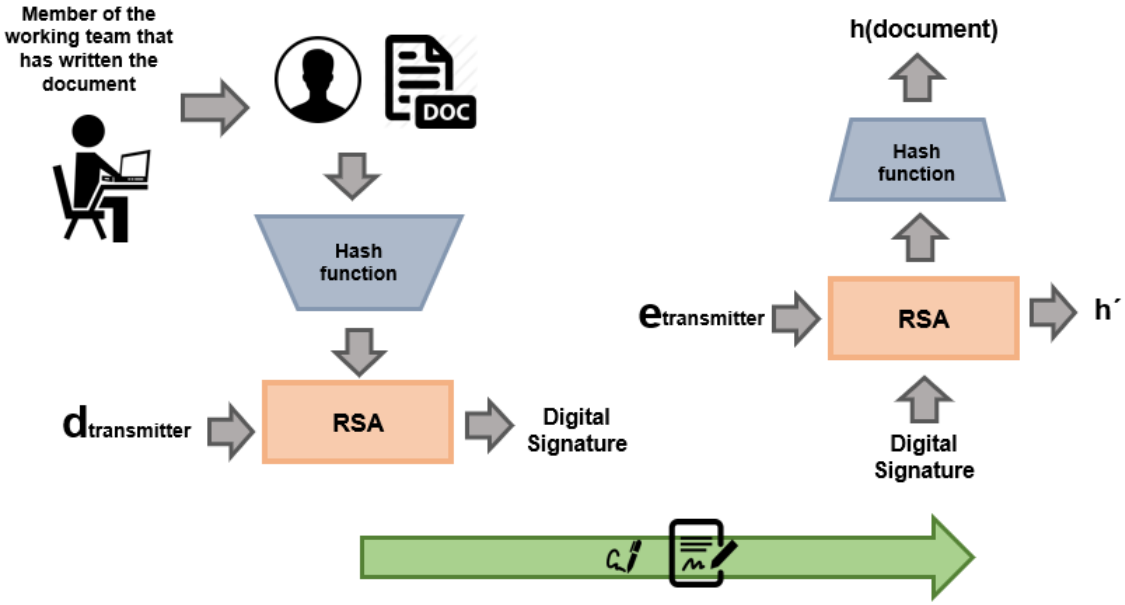
When a minute is created we know that it requires to be signed by each member of the working team. For that reason we have planned that each member of the team will have their own digital sign, so that they use it to sign the document. On the other hand, the minute must be saved in a web platform for it to be seen by everyone that participated in the meeting.

Notably, using digital signature will be a good idea for working with documents that will be saved in a server, using a Hash function we will get a reduction in size.



2.2.1 Memorandum

The difference between a minute and memorandum is that in a memorandum we only need the sign of the member of the working team that has written the document, for that reason we can use the same logic of the previous document, to apply a digital signature. A memorandum, like a minute, has to be available to be seen whenever somebody wants to check it, so we will use the same hash function.



Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

Confidential memorandum

A confidential memorandum shares the same characteristics of a normal memorandum, but only it must only be seen by the sender and the receiver. In order to manage the privacy of this document, we'll use another security service that will be explained in the following section.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

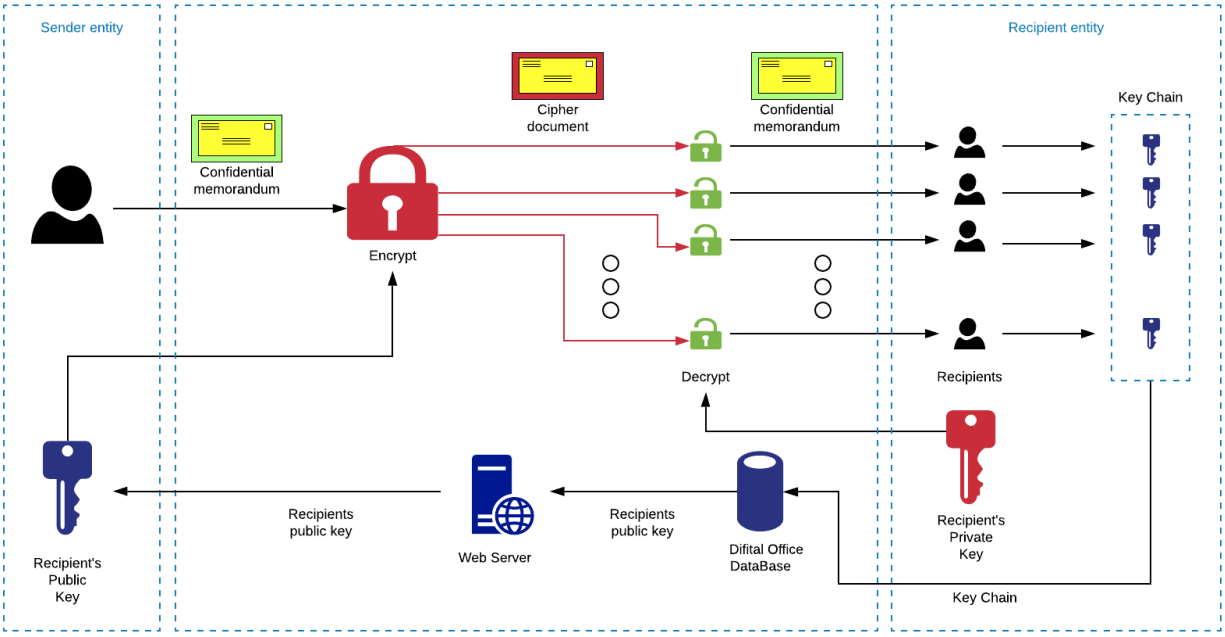
2.3 Secure documents

In the company are handled different documents which are minutes, memorandum and confidential memorandum, of which only the third document is required that is not available to any entity

A confidential memorandum it's similar to a memorandum, but with the important feature that it is only visible by the people authorized by the issuer, so we must concentrate on that this document has special treatment.

In the case of this item, it was identified that it is necessary to provide confidentiality, since we deal with a document that may contain sensitive information about the company.

The cryptographic mechanism that we have planned to use to provide confidentiality in the system is the end to end encryption as shown in the following image



As you can see in the previous image, there is a sender that uses a keychain which is retrieved from the database of the system, where each key corresponds to the public key of each of the entities authorized to receive the confidential memorandum. By making use of the public keys of the recipients, we provide confidentiality to the document.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

CHAPTER 3

Calendar of activities

3.1 Links about Digital Office

- **GitHub**: We have created a GitHub repository to save everything about the project, we will put full documentation and main code. https://github.com/abiisnn/Digital-Office.
- **Google sheets Schedule**: PERT and Gantt diagram, it can be updated. https://docs.google.com/spreadsheets/d/101TKTB6AVobsn1EqXq0sonLwlyFPV2uDGl8ImoijK4c/edit?usp=sharing.

3.2 Google sheets schedule

In this section, we will explain what we are doing to show us calendar of activities. We have sent a google sheets link that allow to show you the PERT and Gantt diagram.

We are using Program Evaluation and Review Technique, PERT is a three point activity estimating technique that considers estimation uncertainty and risk by using three estimates to define an approximate probability for an activity's cost or duration. We have two important sections that are taking in count, **Analysis** and **Developing**.

3.3 Program Evaluation and Review Technique

CHAPTER 4

Identification of requirements

4.1 General process

We use it to indicate the importance for each requirement:

- H - High
- N - Normal
- L - Low

Requerimientos funcionales				
Id	Nombre	Descripción	Prioridad	Origen
RU2	Bill signing	The system must allow every minute to get signed by every participating member in the meeting	H	Origin
RU3	Destinatary verification	The system must verify that every memorandum has the signature of whom wrote the document	H	Origin
RU4	Dissemination of the minutes	The system must allow each member of the organization to view an official written statement.	N	Origen
RU5	Confidential memorandum permission	The system must have the option to grant permissions to the people who can visualize confidential memorandums.	H	Origin
RU6	Register meetings	The system must register the information of all meetings held in the office	N	Origin
RU7	Integrity of the documents	The system will determine if a document was altered without authorization	H	Origin

CHAPTER 5

Process

5.1 Create a minute

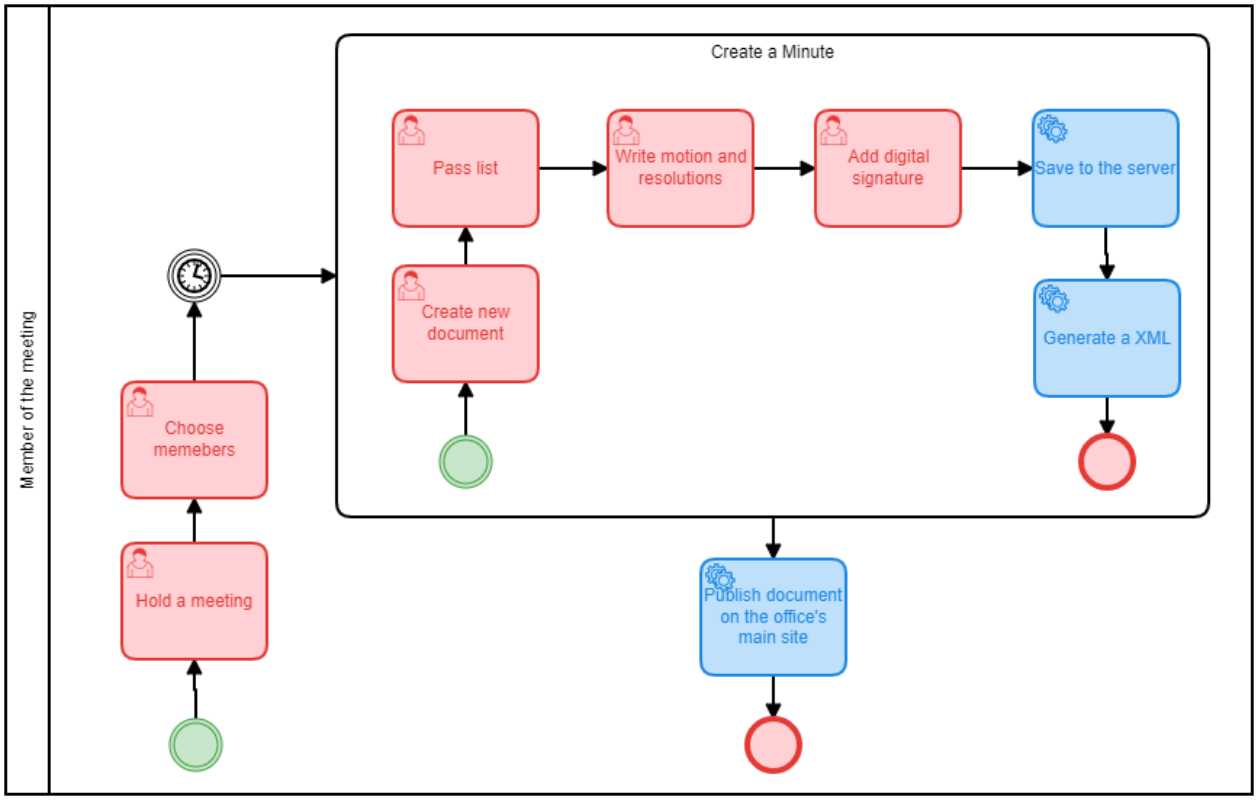


Figure 5.1: Minute process

- **Hold a meeting:**
  The CEO chooses the members present in the meeting and also will choose who will write the motion and resolutions.
- **Choose members:**
  The CEO have to choose the members of the office that will be present in the meeting. We need this part because will help to pass list and also to sign.
- **Create new document**
  When the members of the meeting want to write motion and resolutions, choose the section create new bill.
- **Check in:**
  We need to check in to know the members present in the meeting, to know who will sign the document.
- **Write motion and resolutions:**
  This part is done by the person designated by the CEO. The motions and resolutions are the notes that have a relation with the meeting.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

- **Add digital signature:**
  Each member of the meeting will sign the document. Each member must have their digital signature.
- **Save to the server:**
  The document will be processed in the server.
- **Generate XML:**
  We generate a XML to save the document and the signatures.
- **Publish document on the office's main site:**
  Every member of the office that participated in the meeting can see the bill.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉

5.2 Cryptographic Services

5.2.1 Confidentiality

This cryptographic service is present in our systen when we cipher the confidential memorandum. In this operation when use the public key of each user in order to cipher the document. Whenever an user, that is the receiver of the confidential memorandum, wants to see the content of a confidential memorandum they have to upload their private key to decrypt the memo. The system checks if the key that was uploaded matches with the encryption that used the public key. Here we fulfill confidentiality by encrypting the content and ensuring that only the receiver will be able to check its content.

5.2.2 Authentication

We fulfill this cryptographic service by implementing the digital dignature in every document. Whenever we want to create a memoradum, confidential or public, the creator of the memo has to sign the document with their private key in order to save the document in the database. In the case of the bill, every participating member of the meeting has to sign the bill with their private key.

In this two processes we sign the documents with the private key, by this process of signing we can ensure the authentication of the documents because every user has their own private key and two users will never share the same private key, so that we can ensure that each user signed the document with their own private key. Also, me have to mention that the private key has to be stored by each user in a safe place, so no one will ever gain access to it.

Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizabal S/N Col. Lindavista, GAM, D. F. ☎ 57296000 Ext. 52054 ✉