

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et
De la Recherche Scientifique



Université Saad Dahleb Blida -
1- Faculté des Sciences
Département d'Informatique



MEMOIRE DE FIN D'ETUDES EN
VUE DE L'OBTENTION DU DIPLÔME
DE MASTER EN INFORMATIQUE

Options: Sécurité des systèmes d'informations

Ciphertext-policy Attribute-based encryption

Réalisé par :

- Remini Rami
- Djellouli Mohammed Akram

Promotrice :

- Mme. Ghebghoub Yasmine

Jurés :

- Mme. Aroussi Sana
- Mme .Nasri Ahlem

Remerciement

Before anything chosen, we would like to thank god Almighty, for giving us the strength and patience to complete this modest work.

We express our deepest thanks to Mrs. GHEBGHOUB, for having accepted to supervise us; her valuable advice, guidance and corrections were very profitable for us.

We also thank the head of the department and all the teaching staff of the computer science department and especially the teachers of the Security of Information Systems specialties which have contributed directly or indirectly to our training during the five years we spent El Hamduli'Allah

Dédicace

We dedicate this modest work as a sign of respect, recognition and thanks:

To our dear parents, who never cease to give me love so that I can achieve what we are today. May gods protect them and may success always be within our reach so that we can fill them with happiness

All our relatives All of our friends, colleagues and all who encouraged us.

Abstract

From an information security perspective storing and sharing sensitive data on the cloud can arise a number of questions about the various threats facing the cloud, including user privacy and the integrity of stored data. The objective of this thesis is to implement solutions to these two problems within the framework we proposed. To do this, we have developed two methods, each of which is the subject of a contribution. The first contribution provides an access control model based on the CP ABE attribute-based encryption method, which allows data owners to ensure data security and provide users with fine access to data using defined policies and constraints. The second a digital signature, which is one of the solutions that can be used to preserve the confidentiality of personal data. The experimental results of our solution show that both methods are effective in safely managing data stored in the cloud and ensuring user confidentiality.

Keywords: Cloud computing security, confidentiality, data integrity, CP-ABE encryption, Digital signature, Access control, Hashing.

Resumé

Du point de vue de la sécurité de l'information, le stockage et le partage de données sensibles sur le cloud peuvent soulever un certain nombre de questions sur les diverses menaces auxquelles le cloud est confronté, notamment la confidentialité des utilisateurs et l'intégrité des données stockées. L'objectif de cette thèse est de mettre en œuvre des solutions à ces deux problèmes dans le cadre que nous avons proposé. Pour ce faire, nous avons développé deux méthodes qui font chacune l'objet d'une contribution. La première contribution fournit un modèle de contrôle d'accès basé sur la méthode de cryptage basée sur les attributs CP ABE, qui permet aux propriétaires de données d'assurer la sécurité des données et de fournir aux utilisateurs un accès précis aux données en utilisant des politiques et des contraintes définies. La seconde une signature numérique, qui est l'une des solutions pouvant être utilisées pour préserver la confidentialité des données personnelles. Les résultats expérimentaux de notre solution montrent que les deux méthodes sont efficaces pour gérer en toute sécurité les données stockées dans le cloud et garantir la confidentialité des utilisateurs.

Mots – clés : Sécurité du cloud computing, confidentialité, intégrité de données, Chiffrement CP-ABE, Digital signature, Contrôle d'accès, Hashing.

ملخص

من منظور أمن المعلومات ، يمكن أن يؤدي تخزين البيانات الحساسة ومشاركتها على السحابة إلى طرح عدد من الأسئلة حول التهديدات المختلفة التي تواجه السحابة ، بما في ذلك خصوصية المستخدم وسلامة البيانات المخزنة. الهدف من هذه الرسالة هو تنفيذ حلول لهاتين المشكلتين ضمن الإطار الذي اقترحناه . للقيام بذلك ، قمنا بتطوير طريقتين ، كل منهما موضوع مساهمة. توفر المساهمة الأولى نموذجًا للتحكم في الوصول يعتمد على أسلوب التشفير المستند إلى سمة CP ABE ، والذي يسمح لمالكي البيانات بضمان أمن البيانات وتزويد المستخدمين بوصول جيد إلى البيانات باستخدام سياسات وقيود محددة. والثاني توقيع رقمي وهو أحد الحلول التي يمكن استخدامها للحفاظ على سرية البيانات الشخصية. تظهر النتائج التجريبية لحلنا أن كلا الطريقتين فعالتان في إدارة البيانات المخزنة في السحابة بأمان وضمان سرية المستخدم.

الكلمات – المفتاحية: أمن الحوسبة السحابية ، السرية ، تكامل البيانات ، تشفير CP-ABE ، التوقيع الإلكتروني ، التحكم في الدخول ، hashing.

Content table

Abstract.....	4
Resumé	5
ملخص.....	6
Content table	7
Chapter 1: Information systems' security.....	11
1.1. Introduction:.....	12
1.2. Definition	12
1.3. The Information Systems Security objectives:	13
1.3.1. Confidentiality	13
1.3.2. Integrity	13
1.3.3. Availability	14
1.4. Tools for Information Systems Security.....	15
1.4.1. Authentication.....	15
1.4.2. Access Control	18
1.4.3. Encryption.....	21
1.5. Conclusion	23
Chapter 2: The security mechanisms.....	24
2.1 General Introduction:.....	25
2.2 The digital signature (QR Code authentication):	25
2.2.1 Definition	25
2.2.2 Background on digital signature:	27
2.2.3 QR Code Authentication process:.....	27
2.2.4 The benefits of using the QR Code:	28
2.3 CP-ABE:	28
2.3.1 Introduction.....	28

2.2. Basic Concepts	28
2.4. Access Control	48
2.4.1. Types of access control	48
2.4.6. Benefits of Access Control	51
2.4.7. Choosing the Best Access Control System for Your Organization	52
2.4.8. Knowing When to Use RBAC vs ABAC	52
2.4.9. Attribute-based Access Control (ABAC):	53
Chapter 3: Conception and realization	58
3.1. Implementation	59
3.2. QR Authentication	59
3.3. Encryption and Access control	61
3.4. Software Presentation.....	62
3.4.1. Users platforms.....	65
3.5. Conclusion	69
GENERAL CONCLUSION	70
Bibliography.....	72

Table of Figures

Figure 1: The security triad [2]15

Figure 2: Symmetric encryption.....29

Figure 3: Block encrypt30

Figure 4:EBC encryption mode30

Figure 5:CBC encryption mode31

Figure 6:Stream Encryption.....32

Figure 7:asymmetric encryption.....32

Figure 8: RSA Process[15].34

Figure 9: RC4 Process.36

Figure 10: DES Process.....38

Figure 11: AES Process.....40

Figure 12: ABE Process.....43

Figure 13: KP ABE Process.....44

Figure 14 : CP ABE Process.46

Figure 15:generate QR code image59

Figure 16: read QR code image60

Figure 19Figure 17: key generation61

Figure 18: access verification62

Figure 19: register frame.....63

Figure 20:login frame65

Figure 21:data frame.....66

Figure 22encryption frame.....67

Figure 23: policy tree.....68

Figure 24: user frame.....68

Figure 25:decrypt frame69

Chapter 1: Information systems' security

1.1. Introduction:

Today and due to technology, people have become more connected to each other than ever, perhaps in every field in our daily life. However, on the other side, personal information and data are easily compromised and vulnerable, that is why the security of information systems is one of the most important requirements in this century.

To ensure this protection, we will use access control as the first line of defense against any unauthorized people and attacks then we will use cryptography to ensure internal and external security.

In this first chapter, we are going to focus on the security of information systems, the goals behind it, and the tools necessary to ensure it.

1.2. Definition

Information systems security, more commonly referred to as INFOSEC, refers to the processes and methodologies involved with keeping information confidential, available, and assuring its integrity.

According to the dictionary of Military and Associated Terms of the US Department of Defense, information systems security is “The protection of information and information systems against unauthorized access or modification of information, whether in storage, processing, or transit, and against denial of service to authorized users. Information security includes those measures necessary to detect, document, and counter such threats. Information security is composed of computer security and communications security.” [1]

It also refers to:

- Access controls, which prevent unauthorized personnel from entering or accessing a system.
- Protecting information no matter where that information is, i.e. in transit (such as in an email) or in a storage area.
- The detection and remediation of security breaches, as well as documenting those events.

1.3. The Information Systems Security objectives:

The objectives and the aspects that we are trying to control are:

- Confidentiality
- Integrity
- Availability

1.3.1. Confidentiality

Confidentiality measures protect information from unauthorized access and misuse. Most information systems house information that has some degree of sensitivity. It might be proprietary business information that competitors could use to their advantage, or personal information regarding an organization's employees, customers or clients.

Confidential information often has value and systems are therefore under frequent attack as criminals hunt for vulnerabilities to exploit. Threat vectors include direct attacks such as stealing passwords and capturing network traffic, and more layered attacks such as social engineering and phishing. Not all confidentiality breaches are intentional. A few types of common accidental breaches include emailing sensitive information to the wrong recipient, publishing private data to public web servers, and leaving confidential information displayed on an unattended computer monitor.

There are many countermeasures that organizations put in place to ensure confidentiality. Passwords, access control lists and authentication procedures use software to control access to resources. These access control methods are complemented by the use encryption to protect information that can be accessed despite the controls, such as emails that are in transit. Additional confidentiality countermeasures include administrative solutions such as policies and training, as well as physical controls that prevent people from accessing facilities and equipment. [2]

1.3.2. Integrity

Integrity measures protect information from unauthorized alteration. These measures provide assurance in the accuracy and completeness of data. The need to protect information includes both data that is stored on systems and data that is transmitted between systems such

as email. In maintaining integrity, it is not only necessary to control access at the system level, but to further ensure that system users are only able to alter information that they are legitimately authorized to alter. [2]

As with confidentiality protection, the protection of data integrity extends beyond intentional breaches. Effective integrity countermeasures must also protect against unintentional alteration, such as user errors or data loss that is a result of a system malfunction.

Many countermeasures can be put in place to protect integrity. Access control and rigorous authentication can help prevent authorized users from making unauthorized changes. Hash verifications and digital signatures can help ensure that transactions are authentic and that files have not been modified or corrupted. Equally important to protecting data integrity are administrative controls such as separation of duties and training.

1.3.3. Availability

In order for an information system to be useful it must be available to authorized users. Availability measures protect timely and uninterrupted access to the system. Some of the most fundamental threats to availability are non-malicious in nature and include hardware failures, unscheduled software downtime and network bandwidth issues. Malicious attacks include various forms of sabotage intended to cause harm to an organization by denying users access to the information system. [2]

The availability and responsiveness of a website is a high priority for many business. Disruption of website availability for even a short time can lead to loss of revenue, customer dissatisfaction and reputation damage. The Denial of Service (DoS) attack is a method frequently used by hackers to disrupt web service. In a DoS attack, hackers flood a server with superfluous requests, overwhelming the server and degrading service for legitimate users. Over the years, service providers have developed sophisticated countermeasures for detecting and protecting against DoS attacks, but hackers also continue to gain in sophistication and such attacks remain an ongoing concern.

Availability countermeasures to protect system availability are as far ranging as the threats to availability. Systems that have a high requirement for continuous uptime should have significant hardware redundancy with backup servers and data storage immediately available. For large, enterprise systems it is common to have redundant systems in separate physical

locations. Software tools should be in place to monitor system performance and network traffic. Countermeasures to protect against DoS attacks include firewalls and routers.



Figure 1: The security triad [2]

1.4. Tools for Information Systems Security

In order to ensure the confidentiality, integrity, and availability of information, organizations can choose from a variety of tools. Each of these tools can be utilized as part of an overall information-security policy, which will be discussed in the next section. [3]

1.4.1. Authentication

1.4.1.1. Definition

Authentication is the process of determining whether someone or something is, in fact, who or what it says it is. Authentication technology provides access control for systems by checking to see if a user's credentials match the credentials in a database of authorized users or in a data authentication server. In doing this, authentication assures secure systems, secure processes and enterprise information security.

There are several authentication types. For purposes of user identity, users are typically identified with a user ID, and authentication occurs when the user provides credentials such as a password that matches their user ID. The practice of requiring a user ID and password is known as single-factor authentication (SFA). In recent years, companies have strengthened authentication by asking for additional authentication factors, such as a unique code that is

provided to a user over a mobile device when a sign-on is attempted or a biometric signature, like a facial scan or thumbprint. This is known as two-factor authentication (2FA). [4]

1.4.1.2.The work of Authentication

During authentication, credentials provided by the user are compared to those on file in a database of authorized users' information either on the local operating system server or through an authentication server. If the credentials entered match those on file and the authenticated entity is authorized to use the resource, the user is granted access. User permissions determine which resources the user gains access to and also any other access rights that are linked to the user, such as during which hours the user can access the resource and how much of the resource the user is allowed to consume.

Traditionally, authentication was accomplished by the systems or resources being accessed. For example, a server would authenticate users using its own password system, login IDs, or usernames and passwords.

However, the web's application protocols -- Hypertext Transfer Protocol and HTTP Secure -- are stateless, meaning that strict authentication would require end users to re-authenticate each time they access a resource using HTTPS. To simplify user authentication for web applications, the authenticating system issues a signed authentication token to the end-user application; that token is appended to every request from the client. This means that users do not have to sign on every time they use a web application.

1.4.1.3.Authentication factors

Authenticating a user with a user ID and a password is usually considered the most basic type of authentication, and it depends on the user knowing two pieces of information -- the user ID or username, and the password. Since this type of authentication relies on just one authentication factor, it is a type of SFA.

Strong authentication is a term that is typically used to describe a type of authentication that is more reliable and resistant to attack. Strong authentication typically uses at least two different types of authentication factors and often requires the use of strong passwords containing at least eight characters, a mix of small and capital letters, special symbols and numbers.

An authentication factor represents a piece of data or attribute that can be used to authenticate a user requesting access to a system. An old security adage has it that authentication factors can

be something you know, something you have or something you are. Additional factors have been proposed and put into use in recent years, with location serving in many cases as the fourth factor and time serving as the fifth factor.

Currently used authentication factors include the following:

- **Knowledge factor.** The knowledge factor, or something you know, may be any authentication credentials that consist of information that the user possesses, including a personal identification number (PIN), a username, a password or the answer to a secret question.
- **Possession factor.** The possession factor, or something you have, may be any credential based on items that the user can own and carry with them, including hardware devices, like a security token or a mobile phone used to accept a text message or to run an authentication app that can generate a one-time password (OTP) or PIN.
- **Inherence factor.** The inherence factor, or something you are, is typically based on some form of biometric identification, including fingerprints or thumbprints, facial recognition, retina scan or any other form of biometric data.
- **Location factor.** Where you are may be less specific, but the location factor is sometimes used as an adjunct to the other factors. Location can be determined to reasonable accuracy by devices equipped with the Global Positioning System or with less accuracy by checking network addresses and routes. The location factor cannot usually stand on its own for authentication, but it can supplement the other factors by providing a means of ruling out some requests. For example, it can prevent an attacker located in a remote geographical area from posing as a user who normally logs in only from their home or office in the organization's home country.
- **Time factor.** Like the location factor, the time factor, or when you are authenticating, is not sufficient on its own, but it can be a supplemental mechanism for weeding out attackers who attempt to access a resource at a time when that resource is not available to the authorized user. It may also be used together with location. For example, if the user was last authenticated at noon in the U.S., an attempt to authenticate from Asia one hour later would be rejected based on the combination of time and location.

Despite being used as supplemental authentication factors, user location and current time by themselves are not sufficient, without at least one of the first three factors, to authenticate a user.

1.4.2. Access Control

1.4.2.1.Introduction

Once a user has been authenticated, the next step is to ensure that they can only access the information resources that are appropriate. This is done through the use of access control. Access control determines which users are authorized to read, modify, add, and/or delete information. Several different access control models exist.

...dozens of access control models have been proposed. Only three have achieved success in practice: mandatory access control (MAC, also known as lattice based access control or multilevel security) [22], discretionary access control (DAC) [24], role-based access control (RBAC) [23] and Attribute-based Access Control (ABAC). While DAC and MAC emerged in the early 1970's it took another quarter century for RBAC and ABAC to develop robust foundations and flourish. RBAC and ABAC emerged due to increasing practitioner dissatisfaction with the then dominant DAC and MAC paradigms, inspiring academic research on RBAC. Since then, both RBAC and ABAC have become the dominant form of access control in practice.

1.4.2.2.Access control types:

1.4.2.2.1. Mandatory access control

Mandatory Access Control is the traditional way to define a user access rights. MAC grants access permission through the operating system. It controls the ability of data owners to grant or deny access rights to file system clients. All access control rights are set by the system manager and enforced by the operating system. Customers have no right to modify these access rights. In this model, each file system object has a classification label such as secret level, top secret level, or confidential. Each device and client is assigned a similar classification and clearance level. The operating system checks the credentials of each person or system when accessing a particular resource to determine the access rights of that specific person or device. Even though MAC offers more security for accessing resources, it has a less flexible environment for dealing with access rights.

Mandatory access control

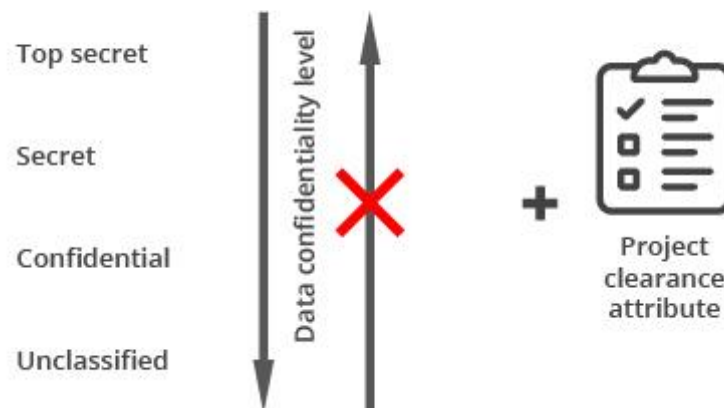


Figure 2: Mandatory access control

1.4.2.2. Discretionary access control

Matrix Access Control or Discretionary Access Control (DAC) is a security access control mechanism that controls access permissions via the data owner. DAC models are discretionary because the owner determines access privileges. In this model, the access rights of each user are defined during authentication by validating the username and password. Additionally, the files or data reside with the owner and the data owner [26] controls the data access policies. The DAC offers more flexibility than the MAC, however, it provides less security.

1.4.2.3. Role based access control

Role-Based Access Control (RBAC) provides access rights based on user roles and privileges. User permissions are defined by various RBAC settings, such as user roles, role permissions, and role relationships.

The notion of role is an enterprise or organizational concept. RBAC allows us to model security from the perspective of the organization, because we can align security modelling to the roles and responsibilities in the organization. Most large organizations have some business rules related to access control policy such as need-to-know, separation of duty, rotation of sensitive job position, and so on. Delegation of authority is an important one of these. Delegation means that a person gives all or part of his authority to somebody. There are three types of situations in which delegation takes place.

1. Backup of role. When an individual is on a business trip or long-term absence, the job functions need to be maintained by others. This requires that somebody be delegated the authority to do the absent individual's job.
2. Decentralization of authority. When an organization needs to setup initially or reorganize subsequently, job functions are distributed from higher job positions to lower job positions in the organization structure.
3. Collaboration of work. Oftentimes people need to collaborate with others in the same organization or other organizations. In this case, we need to grant some access authority to share information.
- 4.

1.4.2.2.4 Attribute based access control

Attribute-based access control (ABAC) is an authentication and authorization model under the identity management umbrella that uses attributes, rather than roles, to grant user access. With ABAC, access decisions are made based on attributes (characteristics) about the subject or user making the access request, the resource being requested, what the user will do with the resource, and the environment (geolocation, network, etc.) or context of the of the request.

ABAC was derived from role-based access control (RBAC), which provides access based on user roles. But while RBAC covers broad access, ABAC can control access on a more detailed level.

Here we will discuss two: the access control list (ACL) and role-based access control (ABAC).

For each information resource that an organization wishes to manage, a list of users who have the ability to take specific actions can be created. This is an access control list, or ACL. For each user, specific capabilities are assigned, such as read, write, delete, or add. Only users with those capabilities are allowed to perform those functions. If a user is not on the list, they have no ability to even know that the information resource exists. [4]

ACLs are simple to understand and maintain. However, they have several drawbacks. The primary drawback is that each information resource is managed separately, so if a security

administrator wanted to add or remove a user to a large set of information resources, it would be quite difficult. And as the number of users and resources increase, ACLs become harder to maintain. This has led to an improved method of access control, called attribute-based access control, or ABAC. With ABAC, instead of giving specific users access rights to an information resource, users are assigned to attribute and then those roles are assigned the access. This allows the administrators to manage users separately, simplifying administration and, by extension, improving security. [4]

1.4.3. Encryption

Many times, an organization needs to transmit information over the Internet or transfer it on external media such as a CD or flash drive. In these cases, even with proper authentication and access control, it is possible for an unauthorized person to get access to the data. Encryption is a process of encoding data upon its transmission or storage so that only authorized individuals can read it. This encoding is accomplished by a computer program, which encodes the plain text that needs to be transmitted; then the recipient receives the cipher text and decodes it (decryption). In order for this to work, the sender and receiver need to agree on the method of encoding so that both parties can communicate properly. Both parties share the encryption key, enabling them to encode and decode each other's messages. This is called symmetric key encryption. This type of encryption is problematic because the key is available in two different places. [1]

An alternative to symmetric key encryption is public key encryption. In public key encryption, two keys are used: a public key and a private key. To send an encrypted message,

you obtain the public key, encode the message, and send it. The recipient then uses the private key to decode it. The public key can be given to anyone who wishes to send the recipient a message. Each user simply needs one private key and one public key in order to secure messages. The private key is necessary in order to decrypt something sent with the public key. [5]

Public Key Encryption Example

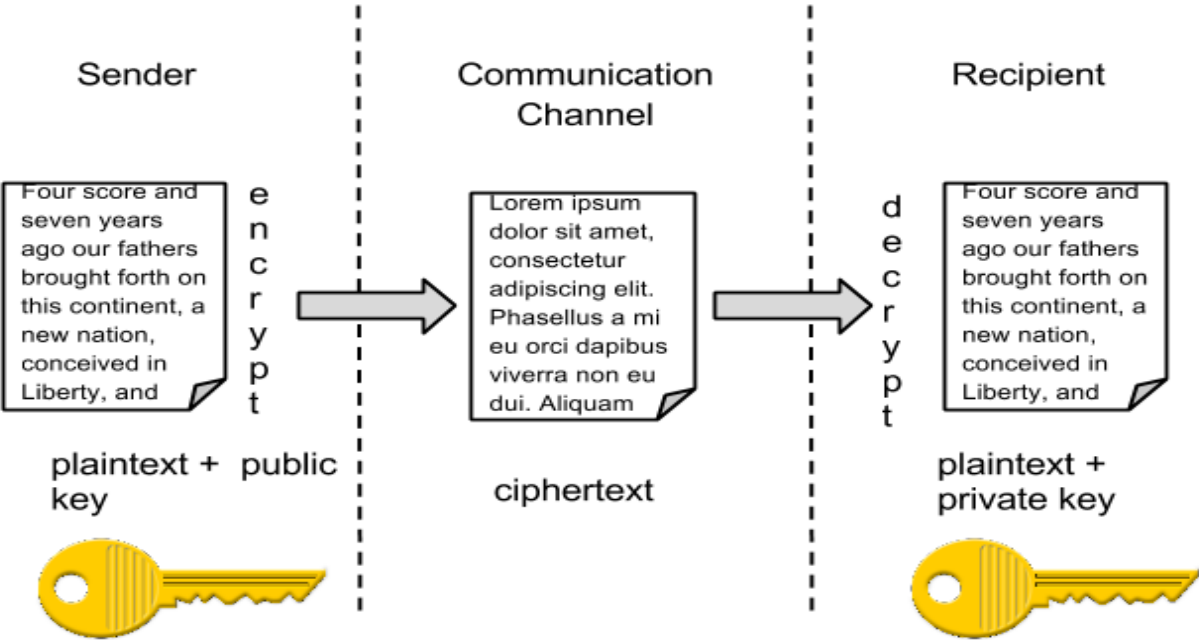


Figure 3:Public Key Encryption Example

1.5. Conclusion

As computing and networking resources have become more and more an integral part of business, they have also become a target of criminals. Organizations must be vigilant with the way they protect their resources. The same holds true for us personally: as digital devices become more and more intertwined with our lives, it becomes crucial for us to understand how to protect ourselves.

Chapter 2: The security mechanisms

2.1 General Introduction:

In this chapter, we will introduce the mechanism used to secure or proposed model.

We will begin by presenting notions of digital signature and QR authentication, then a study of attribute-based encryption and its existing models, and finally we will dive deeper into the attribute-based access control.

2.2 The digital signature (QR Code authentication):

2.2.1 Definition

A QR (“quick response”) code is a two-dimensional barcode. Information is encoded in both the vertical and horizontal direction, thus holding up to several hundred times more data than a traditional bar code. Data is accessed by capturing a photograph of the code and processing the image with a QR reader.

QR Codes consist of different areas that are reserved for specific purposes. Because version 1 does not contain all patterns. Figure 3:

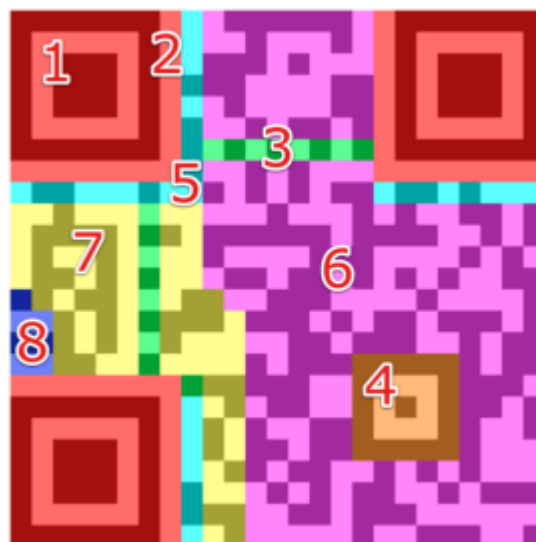


Figure 4: QR Code areas

- **Finder Pattern (1):** The finder pattern consists of three identical structures that are located in all corners of the QR Code except the bottom right one. Each pattern is based on a 3x3 matrix of black modules surrounded by white modules that are again surrounded by black modules. The Finder Patterns enable the decoder software to recognize the QR Code and determine the correct orientation.
- **Separators (2):** The white separators have a width of one pixel and improve the recognizability of the Finder Patterns as they separate them from the actual data.
- **Timing Pattern (3):** Alternating black and white modules in the Timing Pattern enable the decoder software to determine the width of a single module.
- **Alignment Patterns (4):** Alignment Patterns support the decoder software in compensating for moderate image distortions. Version 1 QR Codes do not have Alignment Patterns. With growing size of the code, more Alignment Patterns are added.
- **Format Information (5):** The Formation Information section consists of 15 bits next to the separators and stores information about the error correction level of the QR Code and the chosen masking pattern.
- **Data (6):** Data is converted into a bit stream and then stored in 8 bit parts (called code words) in the data section.
- **Error Correction (7):** Similar to the data section, error correction codes are stored in 8-bit long code words in the error correction section.
- **Remainder Bits (8):** This section consists of empty bits of data and error correction bits cannot be divided into 8-bit code words without remainder.

The entire QR Code has to be surrounded by the so-called Quiet Zone, an area in the same color shade as white modules, to improve code recognition by the decoder software

2.2.2 Background on digital signature:

A digital signature is a bit pattern that depends on the message being signed and uses some information unique to the signer. The message M is fed into a cryptographic hash function resulting in a hash value h or a message digest.

The hash value h that depends on the message M is encrypted using the signer's private key producing the signature. To verify whether or not the digital signature is valid, the result hash value from the message M' is compared to the value from decrypting the signature using the signer's public key. If both values are identical, the owner of the public key is the author of the message. Otherwise, the signature is invalid.

Digital Signature Standard (DSS) includes three techniques, namely; the Digital Signature Algorithm (DSA), the RSA digital signature algorithm [6], and the Elliptic Curve Digital Signature Algorithm (ECDSA). The security of the digital signature depends on the cryptographic hash function and the public key cryptographic algorithm.

For breaking a digital signature, an attacker may create a fraudulent digital signature by creating a new message for an existing digital signature, which is an attack on the cryptographic hash function, or by constructing a fraudulent digital signature for a given message, that is an attack on the public key cryptographic algorithm.

The hash function must be collision resistant and the public key algorithm must be strong against attacks. The approved techniques are considered secure.

It is computationally infeasible to forge a digital signature. The digital signature provides authentication and non-repudiation. Therefore, if the signature is valid, the author of the message cannot deny creating the message

2.2.3 QR Code Authentication process:

The QR code is used in place of both the username, the password and the attributes.

- a- The user insert (upload) the QR Code
- b- The system reads (decipher) the selected code.
- c- The system matches the inserted data with database
- d- The system will load the main interface depending on the user data

2.2.4 The benefits of using the QR Code:

2.3 CP-ABE:

2.3.1 Introduction

Among the main properties of medical data security sought are: reliability, integrity, confidentiality.

To ensure data security and for confidentiality purposes data must be encrypted.

In this chapter we will present notions about cryptography, methods and some data encryption algorithms.

2.1.1. Definition

Cryptography comes from the ancient Greek words *kryptos* (hidden) and *graphein* (to write) and means "secret writing". Its purpose was to protect a secret message during its transmission.

It is made up of three fundamental characteristics [6]:

- Confidentiality which guarantees the secrecy of the message transmitted;
- Integrity which ensures that the message transmitted has not been modified;
- Authentication which verifies the identity of the issuer.

2.2. Basic Concepts

There are mainly two types of cryptography:

- Symmetric encryption : Private key encryption.
- asymmetric encryption : Public Key encryption.

2.2.1. symmetric encryption

In a symmetric cipher or secret key cipher, a sender and a receiver share the same secret key. This key is used for both encryption and decryption and should be kept secret from all other people. This operation is shown in the following figure:

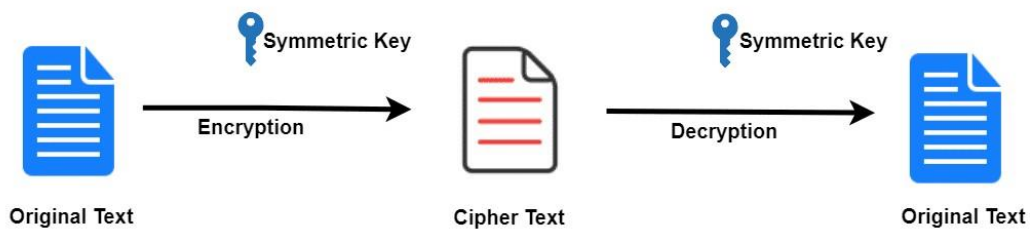


Figure 5: Symmetric encryption

The characteristics of symmetric cryptography are [7]:

- The keys are identical: $KE = KD = K$,
- The key must remain secret,
- The most widespread algorithms are DES, AES, 3DES, ...
- When generating the key, it is chosen randomly,
- These algorithms are based on operations of transposition and substitution of the bits of the plain text according to the key,
- The size of the keys is often of the order of 128 bits. DES uses 56, but AES can go up to 256,
- The main advantage of this encryption mode is its speed,
- The main disadvantage lies in the distribution of keys: for better security, manual exchange is preferred. Unfortunately, for large systems, the number of keys can become large. This is why we will often use secure exchanges to transmit the keys. Indeed, for a system with N users, there will be $N * (N - 1) / 2$ pairs of keys.

2.2.1.1. Block Cipher

A Block Cipher algorithm transforms blocks of data of fixed size into a block of encrypted data of the same size. The blocks are generally 128 bits long, but they can range from 32 to 256 bits depending on the algorithm. The transformation remains the same for each block. Some examples of encryption algorithms: DES, AES, ...

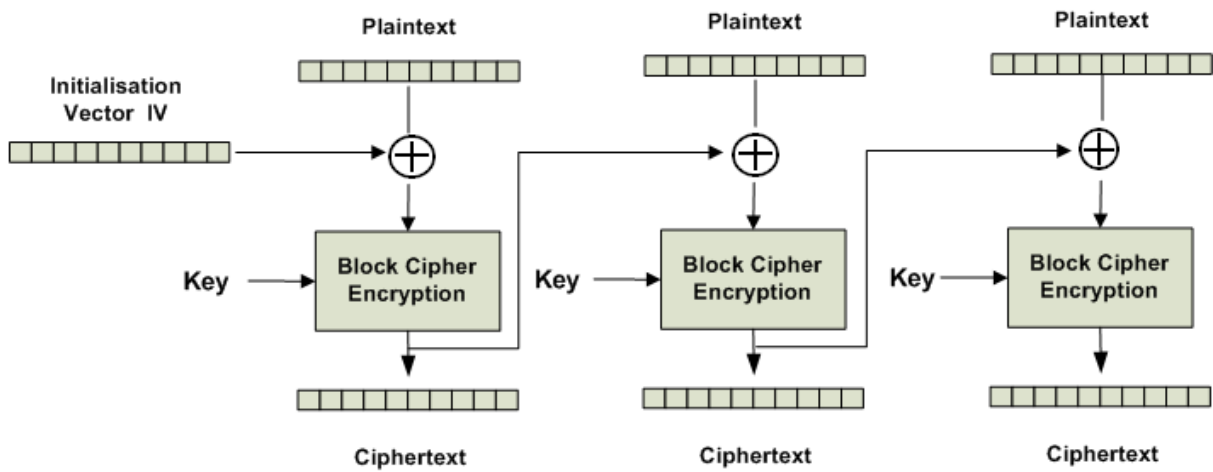


Figure 6: Block encryption

Block cipher can have two encrypting modes which are :

- ECB Mode(*Electronic Code Book*).
- CBC Mode (*Chipher Block Chaining*).

ECB Mode:

This mode allows parallel encryption of the different blocks making up a message. Same plain message block will always be encrypted as one encrypted message block. Significant error propagation; if any bit of the encrypted message is changed during transfer, the entire corresponding plaintext message block will be false [8].

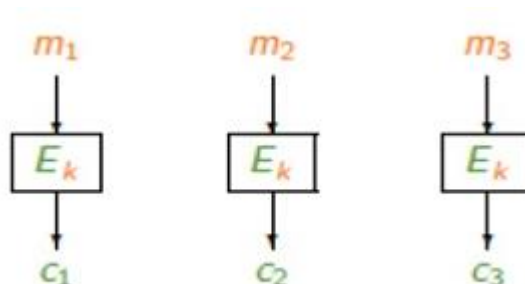


Figure 7:ECB encryption mode

For encryption every plain block m_i is encoded individually and give a new cipher block c_i by the encrypting function E .

CBC Mode:

The structure of the plaintext message is masked by the chaining. An attacker can no longer manipulate the cryptogram, except by removing blocks at the beginning or at the end. It is no longer possible to parallelize the encryption of the different blocks [9].

Block chaining does not cause significant error propagation; if a bit of the encrypted message is changed during transfer, only the corresponding plaintext message block and one bit of the following plaintext message block will be damaged [8]:

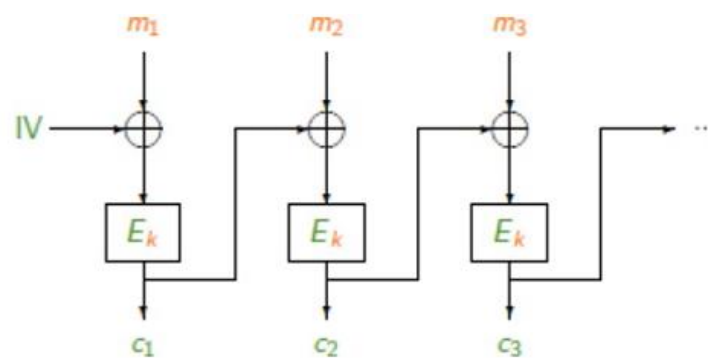


Figure 8: CBC encryption mode

An initialization vector IV is randomly generated $C_i = E_k (M_i \oplus C_{i-1})$. Vector IV is transmitted with the cipher blocks.

2.2.1.2. Stream Cipher

The principle is to generate a pseudo-random stream and combine it with bit-by-bit information by the XOR operation. At reception, we apply the same mechanism, and we return the information. Some examples on encryption algorithms: RC4, E0,...

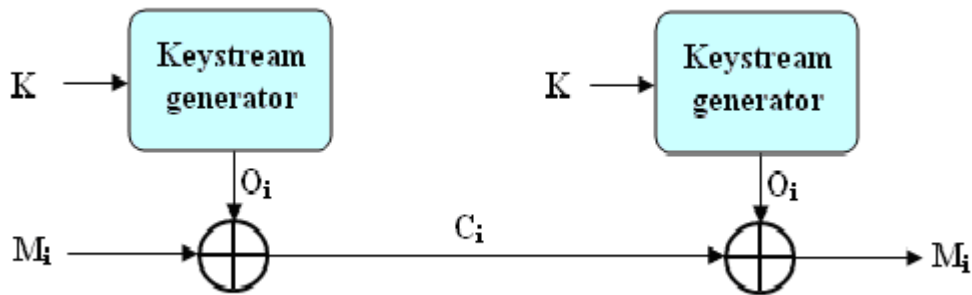


Figure 9:Stream Encryption.

2.2.2. asymmetric encryption

The general idea of asymmetric cryptography (also called public key encryption) is an encryption method that is opposed to symmetric cryptography. It is based on the use of keys, a public key (which is broadcast) and a private key (kept secret), the first allowing to encode the message and the second for the decoder, as shown in the figure below [10].

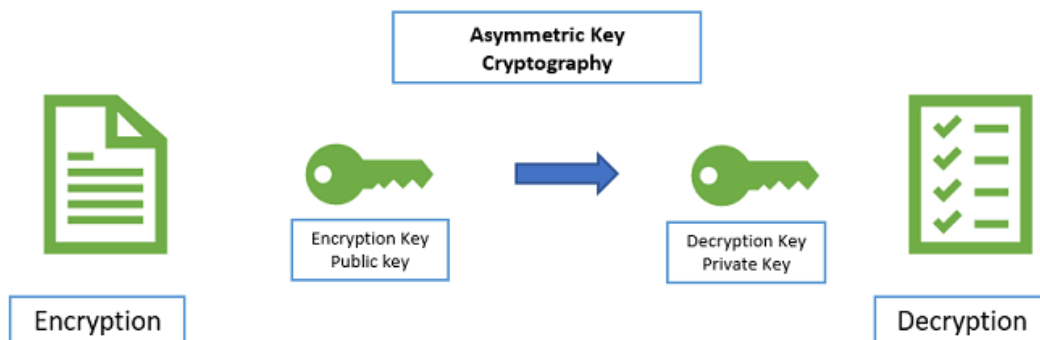


Figure 10:asymmetric encryption.

Asymmetric algorithms have two operating modes [8]:

- Encryption mode: The sender encrypted with the receiver's public key, the receiver decrypts with his private key. In this mode, the sender is sure that only the receiver can decrypt the file.
- Signature mode: The sender signed with his private key, the receiver of the signature with the sender's public key. In this mode, the receiver is sure that it is the sender who sent the file.

The main advantage of public key encryption is to solve the problem of sending private key over an unsecured network. Although slower than most private key encryption, it is still preferable for 3 reasons :

- More scalable for systems with millions of users.
- More flexible authentication.
- Supports digital signatures.

2.2.3. Comparing

	Advantages	disadvantage
Symetric	Encryption and decryption doesn't take a too much time	Low security and very vulnerable to cryptanalysis
Asymetric	Strengthens security, Even by intercepting the message, it cannot be decrypted without the private key	Very complexe algorithms to implement and slow to decrypt and encrypt

2.3. Encryption Algorithms

2.3.1. RSA

RSA:

RSA operations can be decomposed in three broad steps; key generation, encryption and decryption.

Key Generation Procedure:

1. Choose two distinct large random prime numbers p & q such that $p \neq q$.

2. Compute $n = p \times q$.
3. Calculate: $\phi(n) = (p-1)(q-1)$.
4. Choose an integer e such that $1 < e < \phi(n)$
5. Compute d to satisfy the congruence relation $d \times e = 1 \pmod{\phi(n)}$; d is kept as private key exponent.
6. The public key is (n, e) and the private key is (n, d) . Keep all the values d, p, q and ϕ secret.

Encryption Plaintext: $P < n$ Ciphertext: $C = P^e \pmod{n}$.

Decryption Ciphertext: C Plaintext: $P = C^d \pmod{n}$ [11].

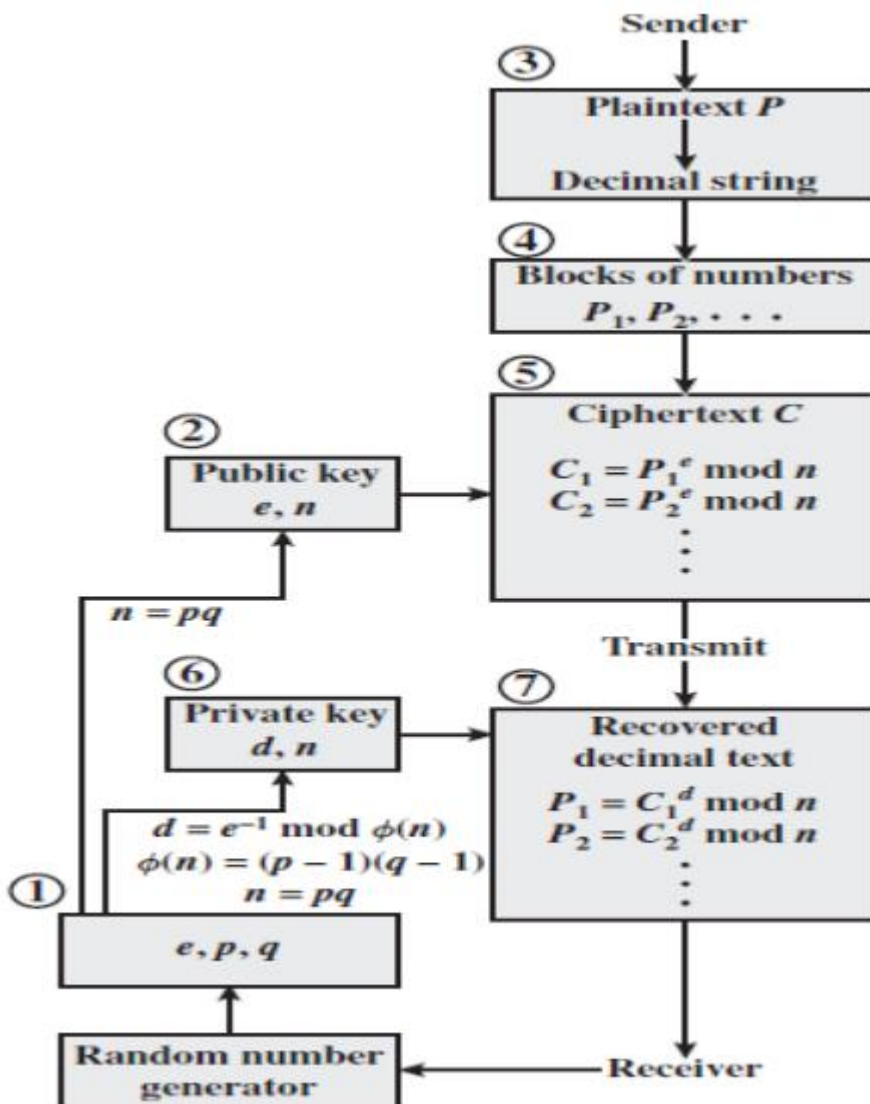


Figure 11: RSA Process[15].

2.3.2. DSA

A Digital Signature Algorithm (DSA) is a public key encoding algorithm established to secure the privacy of numeral text.

The DSA was founded by NIST. A text is signed by a secret key to invent a signature and the signature is checked opposite to the text by a public key. Likewise, any group can check the authenticity signatures; however, only the party with the secret key could sign the texts. An available numeral signature offers a recipient a cause to think that the message was invented by a known sender who has the secret key, and that it was not modulated in transferring [4].

2.3.3. RC4

RC4 is an output-feedback mode cipher. Its keys are 2048 bits long, and its internal state consists of two counters i and j (each within 0 255) plus an array of 256 8-bit bytes, called the S-box. If a cryptography system wishes to use a shorter key, the key is repeated as many times as necessary to fill the 2048-bit key. Once the S-box is initialized with the key, the RC4 algorithm is a loop that updates the internal state of the S-box and returns a byte of keystream. Normally, this keystream is XOR-ed with the plaintext message to produce the ciphertext. As with any output-feedback cipher, RC4 only protects the secrecy of a message, not its integrity. Other measures, such as the use of cryptographic checksums, are commonly used along with RC4.

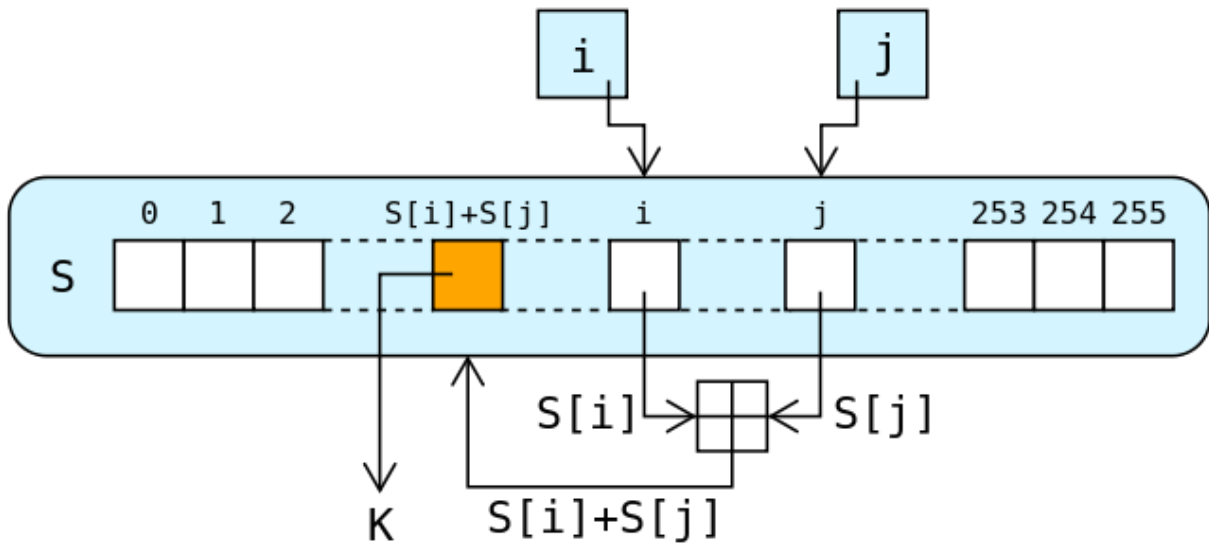


Figure 12: RC4 Process.

RC4 boasts a number of advantages compared to other stream ciphers:

- RC4 is extremely simple to use, thus making the implementation simple as well.
- RC4 is fast, due to its simplicity, which makes it a better performing cipher.
- RC4 also works with large streams of data swiftly and easily.

Though it has advantages, RC4 has many disadvantages as well:

- The vulnerabilities found in RC4 means RC4 is extremely insecure, so very few applications use it now.
- RC4 cannot be used on smaller streams of data, so its usage is more niche than other stream ciphers.
- RC4 also does not provide authentication, so a Man in the Middle attack could occur, and the RC4 cipher user would be none the wiser.

2.3.4. DES

DES encryption instructions are:

- DES receives data of 64-bit long message and 56 bit key and comes up with 64-bit block.
- The ordinary text block needs to modulate the bits.

- The 8 similar bits are eliminated from the key through exposing the key to its key permutation.

The readable message and the key will be produced as the following steps show[4] :

- The key is divided in to two 28 halves.
- The half is rotated by one or two bits, according to the round.
- The two parts reunite and undergo to the round permutation to decrease the key from 56 bits to 48 bits. These pressed keys are used to encode the round's plaintext block.
- The shifted key parts from tip 2 are used in the coming round.
- The database block divides into two 32-bit parts.
- A part will be expanded in terms of permutation to raise the size to 48 bits.
- Result of the sixth step is for OR'ed only, with 48 bit key from tip number three.
- The outcome of 7th instruction is set s-box, that replaces key bits and cut down the 48-bit block to 32 bits
- The consequence of the 8th tip, will be permuted by p-box.
- The result of the p-box belongs to OR'ed solely, will the next part of the format block. The bipartite format parts are exchanged and foerservoir of the coming stage.

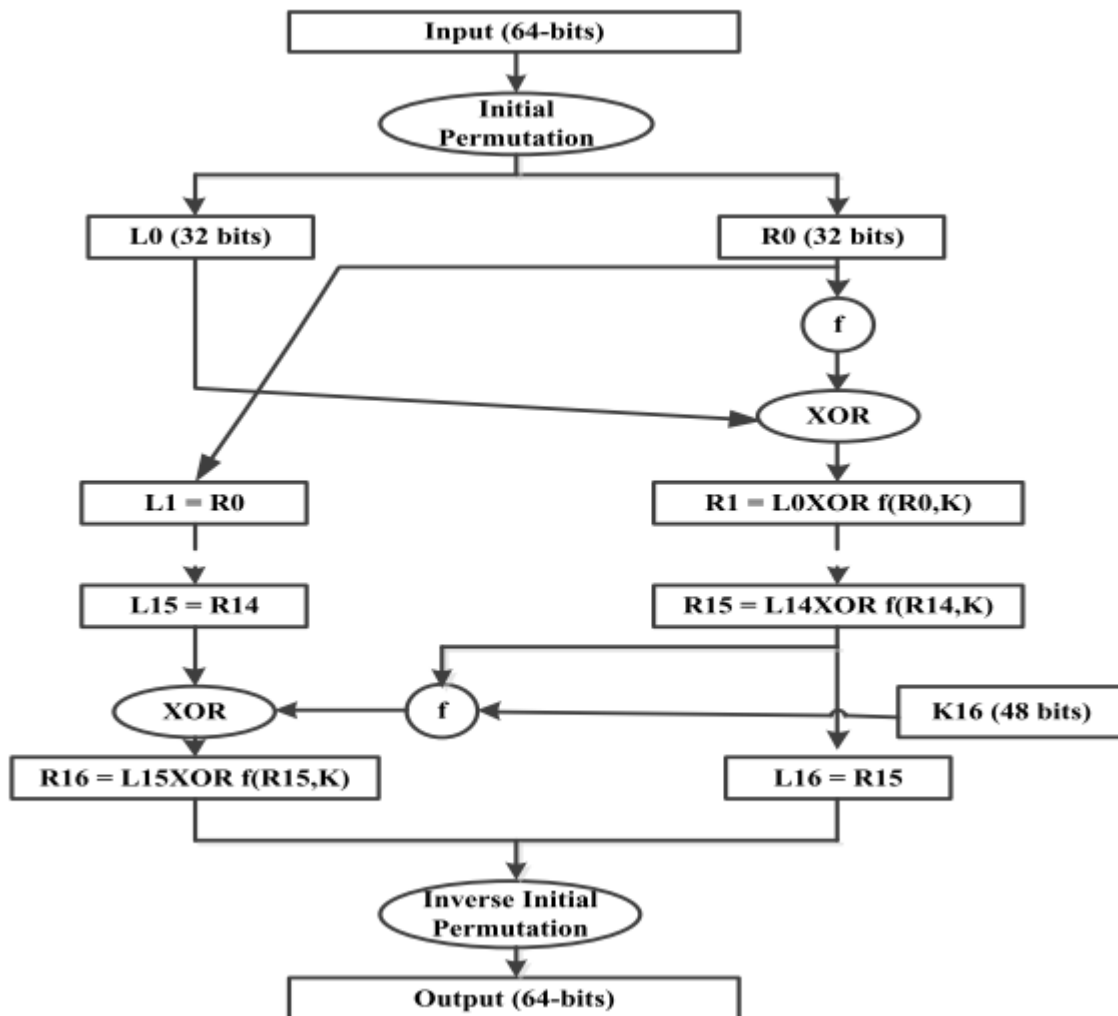


Figure 13: DES Process.

2.3.5. Triple DES

Triple DES (3DES) :

The Triple Data Encryption Algorithm (TDEA or 3DES) was developed to tackle the flaws in DES whilst preserving the same cryptography. 3DES key size of DES (56-bit). This is through implementing the algorithm triple successively with 3 multiple keys. The total size is 168 bits. TDEA uses triple 64-bit DEA keys (K1, K2, K3) in the encode-decode-encode (EDE) state. The standards define three major choices [8]:

- The 1st choice is the preferred one ($K1 \neq K2 \neq K3 \neq K1$).
- The 2nd choice uses dual independent keys ($K1 \neq K2 \& K3 \neq K1$).
- The 3rd choice uses triple similar keys ($K1 = K2 = K3$).

Those choices are equivalent to DES Algorithm. In 3DES, the 3-times iteration is applied to increase the encoded level and average time. It is a known fact that 3DES is slower than other block cipher methods.

2.3.5. AES

During encryption-decryption, the AES process encodes 10 rounds for 128-bit keys. 12 rounds for 192-bit keys and 14 rounds to 256-bit keys to come out with the last encoded message, the algorithm is referred to as AES-128, AES-192, or AES-256, depending on the key length.

AES allows a 128 bit data length that can be divided into four basic operational blocks. These blocks are treated as array of bytes and organized as a matrix of the order of 4×4 that is called the state.

For both encryption and decryption, the cipher begins with an AddRoundKey stage.

However, before reaching the final round, this output goes through nine main rounds, during each of those rounds four transformations are performed: Sub-bytes, Shiftrows, Mix-columns, Add round Key. In the final (10th) round, there is no Mix-column transformation.

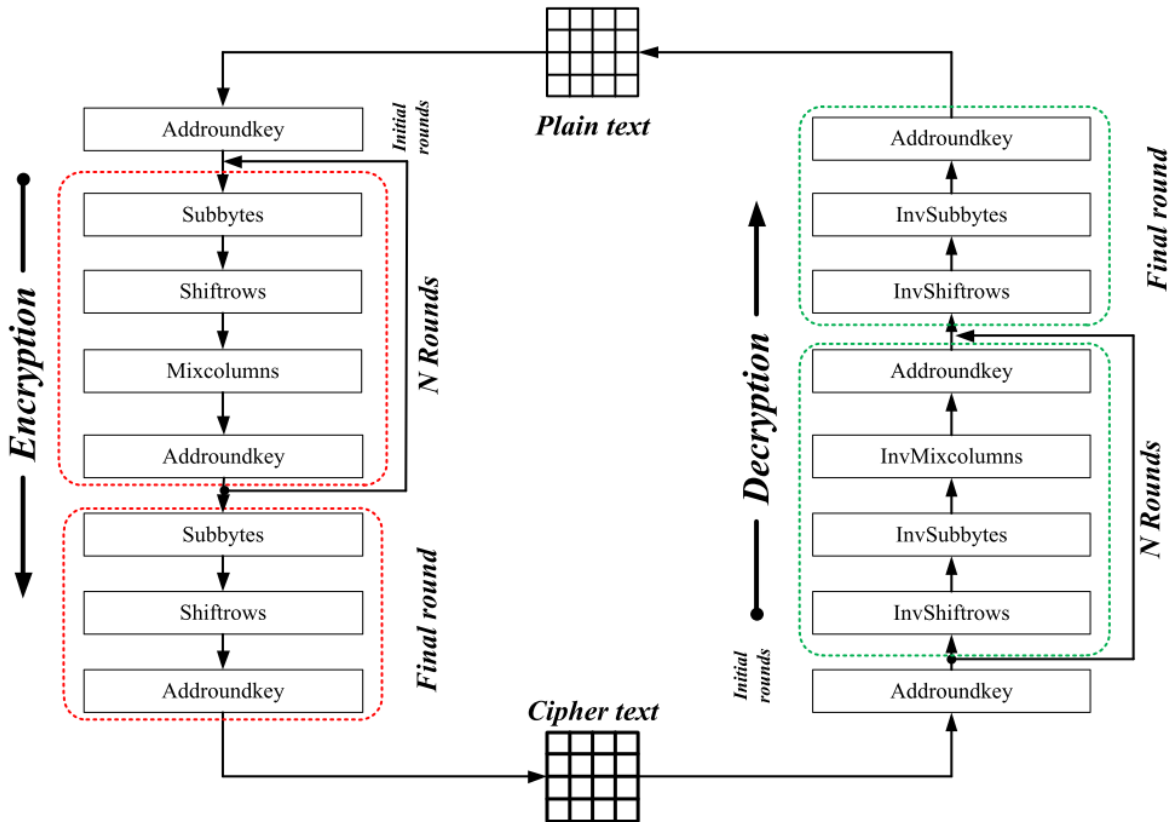


Figure 14: AES Process.

Decryption is the reverse process of encryption and using inverse functions: Inverse Substitute Bytes, Inverse Shift Rows and Inverse Mix Columns [11].

Substitute Byte: AES contains 128 bit data block, which means each of the data blocks has 16 bytes. In sub-byte transformation, each byte (8-bit) of a data block is transformed into another block using an 8-bit substitution box which is known as Rijndael Sbox.

Shift Rows: It is a simple byte transposition, the bytes in the last three rows of the state, depending upon the row location, are cyclically shifted. For 2nd row, 1 byte circular left shift is performed. For the 3rd and 4th row 2-byte and 3-byte left circular left shifts are performed respectively.

Mixcolumns: This round is equivalent to a matrix multiplication of each Column of the states. A fix matrix is multiplied to each column vector. In this operation the bytes are taken as polynomials rather than numbers.

Addroundkey: It is a bitwise XOR between the 128 bits of present state and 128 bits of the round key. This transformation is its own inverse.

2.3.6. Blowfish Algorithm

This algorithm is an encryption method which has been based on a cipher key and a fast and symmetric block cipher. It is easily implemented on fast 32bit processors and needs less than 5k of memory.

Blowfish presents a good encryption rate in software and to this date, no effective cryptanalysis has been discovered on it.

The length of the block used in this algorithm is 64bits while the length of the key is variable in this algorithm and can be from 3 to 448 bits.

This algorithm consists of two sections: key expansion and data encryption. Key expansion transforms a key of variable length (56 bytes maximum) to an array of several subkeys with an overall size of 4168 bytes. The encryption step has 16 rounds.

Each round consists of a permutation, based on the key and another permutation, based on the key and data.

This algorithm, by accepting a public key with a length of 32 to 448 is much faster than other methods like DES and is a good alternative. Also, by having a variable key length and increasing it, brute force attack is impossible on this algorithm [12].

2.4. Attribute based encryption

2.4.1. Background on Identity Base Encryption

Asymmetric encryption has solved the key exchange problem, however, identity-based encryption or IBE (Identity Based-Encryption) [7] solves the problem of public key certification.

Within the framework of this encryption, a user can publicly derive an encryption key for a given identity, for example by an e-mail address. The user with the email address obtains from an authority the secret key to decrypt his identity.

Its principle is to take the identity of the user as a public key, for example his name, first name, date of birth, or his social number. Then create secret encryption keys relating to these identities so that two different individuals cannot have the same secret key, then it is no longer useful to certify the public keys.

The constraint of this approach is that the level of trust that is granted to the private key generator must be very high, because it is intrinsically capable of regenerating the private key of any user, and therefore of being able to perform signatures or decryption without authorization.

2.4.2. ABE Algorithm

Attribute-based encryption (ABE) is one of the advanced cryptographic techniques for one-to-many encryption that overcomes the limited functionalities of the traditional public key cryptographic techniques.

To apply ABE, a data owner encrypts its data using a symmetric encryption algorithm with asymmetric key and then encrypts the key using an ABE scheme with a public key.

The encrypted key is distributed to a group of recipients/users as a ciphertext. Each user obtains the private key for the encrypted key decryption from a key generator that calculates the key

according to the user's attributes. In this case, the data owner does not need to know the identities of the legitimate users and their dynamicity.

Figure 11 illustrates the above-mentioned operational process.

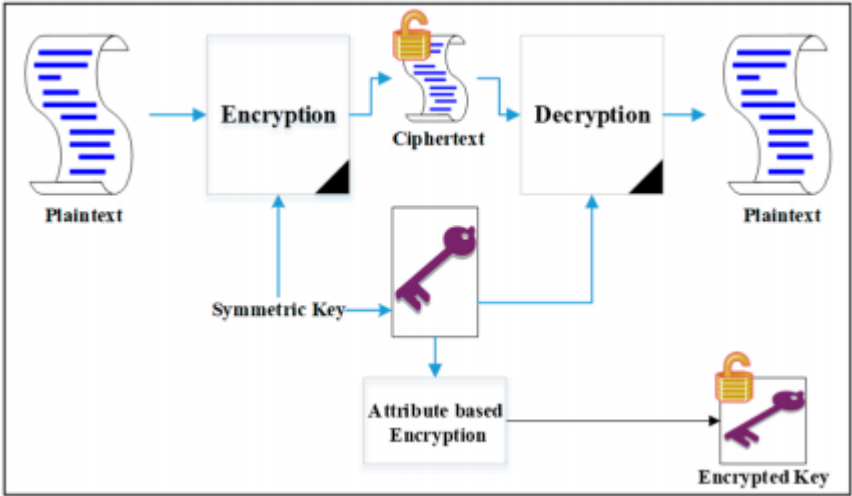


Figure 15: ABE Process.

2.4.3. KP ABE

In a KP-ABE scheme , every ciphertext is associated with a set of attributes, and every user's secret key is associated with an access structure on attributes.

A user will be able to decrypt a ciphertext only if the access structure associated with the user's secret key is satisfied by the set of attributes associated with the ciphertext. This access control functionality can be very powerful, but also costly [5].

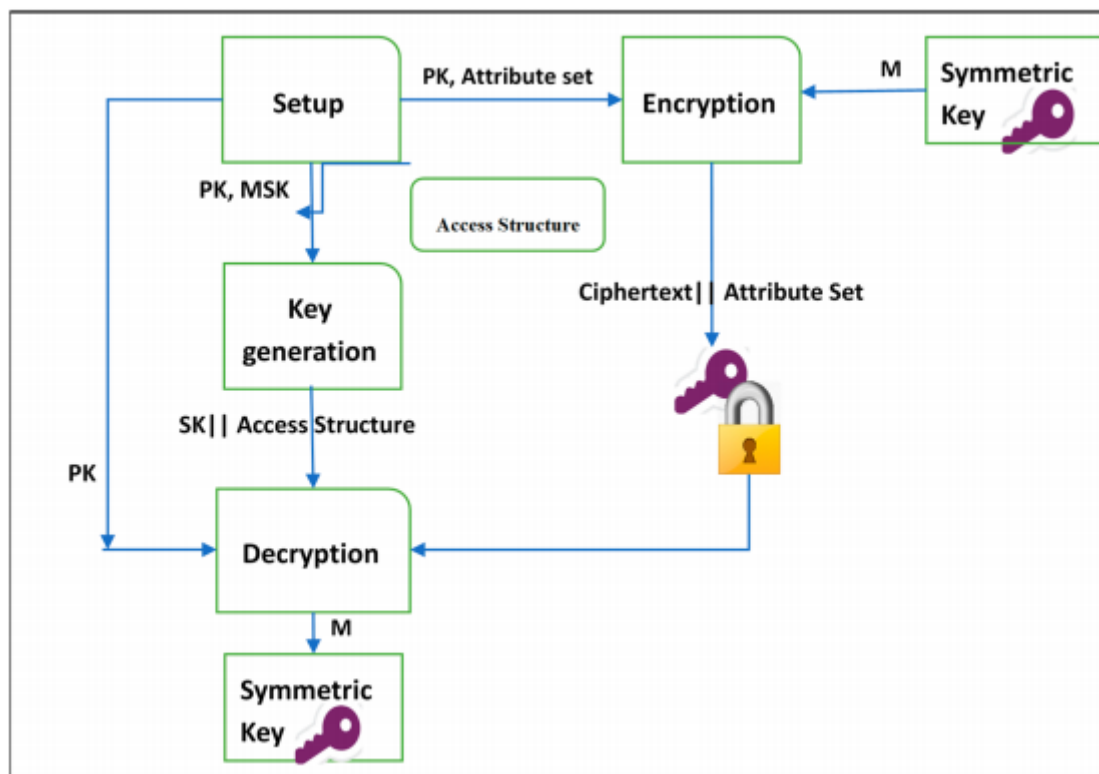


Figure 16: KP ABE Process.

A KP-ABE scheme consists of the following four algorithms:

$\text{Setup}(1\lambda)$ takes as input a security parameter λ . It outputs the public parameters PK and a master secret key MSK .

$\text{KeyGen}(PK, MSK, A)$ takes as input the public parameters PK , the master secret key MSK and an access structure A . It outputs a private key SKA corresponding to A .

$\text{Encrypt}(PK, M, S)$ takes as input the public parameters PK , a message M and a set of attributes S . It outputs a ciphertext CT .

$\text{Decrypt}(PK, SKA, CT)$ takes as input the public parameters PK , a private key SKA , and a ciphertext CT associated with a set of attributes S . If the set S of attributes satisfies the access structure A , then the algorithm will decrypt the ciphertext and return a message M . [13]

2.4.4. CP ABE

The most popular variant of ABE techniques is CP-ABE. Four entities are responsible for running this scheme. These entities are attribute authority, data owner, data user, and cloud server. The role of the attribute authority is to generate secret keys for users according to their attributes to decrypt data. In addition, it is responsible for generating a public key and a master key. The data owner's role is to define an access policy that describes who can access to its data as well as encrypting those data under this access policy. Firstly, a data owner uses a symmetric encryption technique (e.g., AES) to encrypt its data. After that, the owner encrypts the symmetric key under its access policy using CP-ABE by selecting a random value as a secret which is shared using the linear secret sharing scheme technique to generate some values associated with each corresponding attribute in the ciphertext according to the owner's access policy. This policy is determined over a set of attributes by the data owner, and can be demonstrated as a Boolean function with (AND, OR) gates between attributes (e.g., (lecturer AND experience \geq 2 years) OR Professor). Then the encrypted data is sent to the designated cloud for storage including the data ciphertext, the CP-ABE ciphertext and the access policy. Associating the access policy with the ciphertext means that the ciphertext chooses which key can recover the plaintext, giving the data owner more control of its outsourced data. The eligible users who possess the required attributes in a right combination (i.e., satisfy the access policy) can successfully decrypt the encrypted data. As a result, the main benefit from CP-ABE is that sensitive data can be stored on an untrusted server without performing authentication checks for the data access [5].

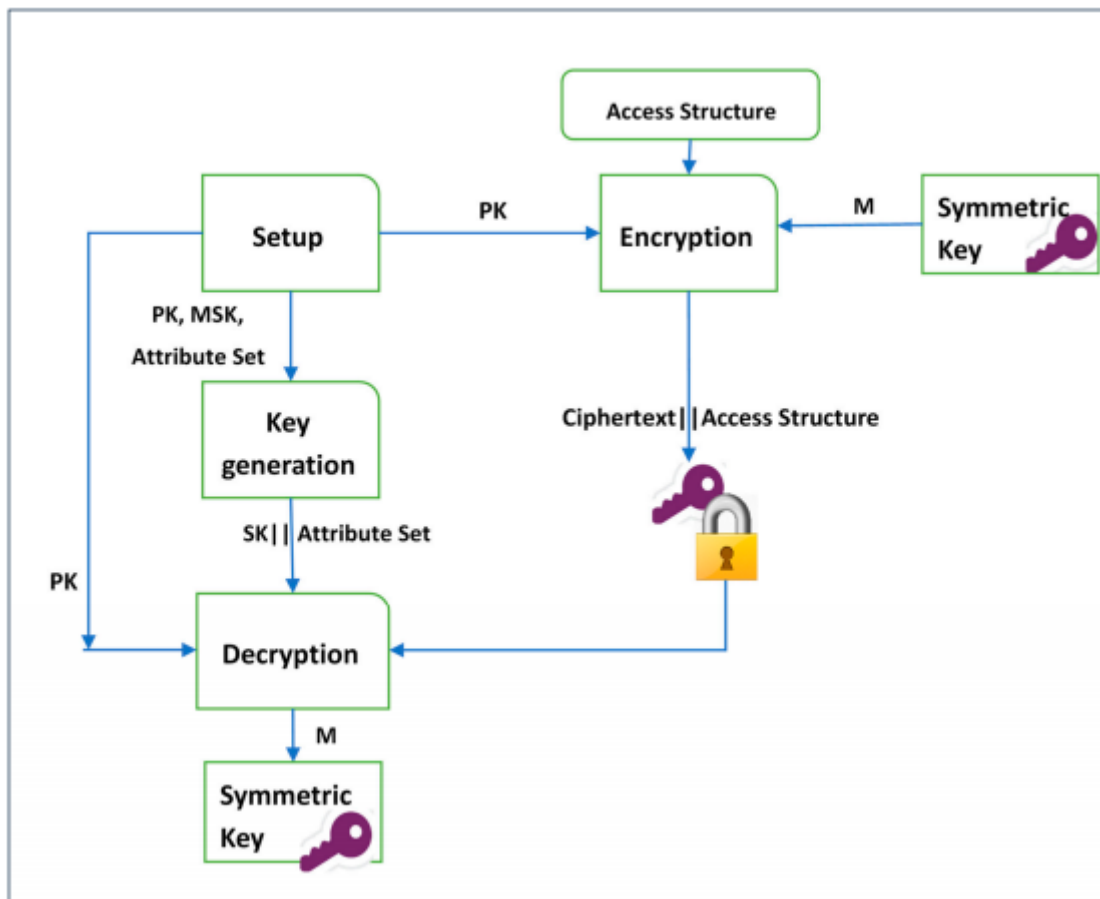


Figure 17 : CP ABE Process.

A common framework of a CP-ABE scheme includes four algorithms, as demonstrated in Figure *: Setup, Encryption, Key Generation, and Decryption [10], which are defined below.

- $\text{Setup}(\lambda, U) \rightarrow (MSK, PK)$: Takes a set of attributes U in the system and an implicit security parameter λ (such as the type of the elliptic curve group used and the base finite field) as inputs to generate a public key PK and a master key MSK as outputs.
- $\text{Encrypt}(PK, A, M) \rightarrow CT$: Takes as inputs a public key PK , an access structure A , and a message M to be encrypted. The output will be a ciphertext CT .
- $\text{KeyGen}(MSK, S) \rightarrow SK$: In this algorithm, a master key MSK and a set of attributes S are taken as inputs. A user's secret key SK is generated as output.
- $\text{Decrypt}(CT, SK) \rightarrow M$: This algorithm takes as inputs a user's secret key SK and a ciphertext CT . It returns a message M when the user's attributes satisfy the access structure.

2.5. Conclusion

In this chapter we have studied the different techniques of cryptography then we have given some examples of data encryption algorithms and at the end we have detailed the technique of attribute encryption.

Encryption appeared in order to strengthen security. ABE encryption features are attractive for a solution that ensures the protection of patient privacy.

2.4. Access Control

Access control is a security measure that controls what and who can access to our system's resources by giving access when it is needed and denying it when it is not. Access control tools help accomplish this purpose, as do Firewalls, encryption, and intrusion detection systems, it is a primary and fundamental concept of security. **Source spécifiée non valide.**

Access Control system (ACS) is based on taking users identifiers such as passwords, personal identification numbers (PINs), biometric scans, tokens or other factors and uses them to verify that the users are who or what they claim to be and authenticates them before granting or denying access to the specific resources. [14]

2.4.1. Types of access control

There are several types of access control in the field and here are some of the main and the most popular ones [14]:

2.4.1. Mandatory Access Control (MAC):

The mandatory access control mechanism (MAC: Mandatory Access Control) is the traditional mechanism for defining user access rights. MAC gives access authorization via the operating system. It controls the ability of owners of data to grant or deny access rights to clients of the file system. All access control rights are defined by the system manager and imposed by the operating system. Customers have no right to modify these access rights. In this model, each object in the filesystem has a classification label such as secret level, top secret or confidential level.

Each device and each customer is assigned to a level of classification and clearance. The operating system verifies the identification informations of each person or system when accessing a particular resource to determine the access rights of that person or that specific device. Even if MAC offers more security for accessing resources, it has a less flexible environment for dealing with access rights. [14]

2.4.2. Discretionary Access Control (DAC):

Discretionary Access Control) is a security access control mechanism that controls access permissions through the data owner.

DAC is an identity-based strategy built-in in most OS that relies on the discretion of the user and the resource owner. In DAC the control is maintained and enforced by an Access Control List (ACL) that is attached to a resource which is defined by its owner to define the access type (grant or deny) to the users of that systems. As it also allows the transfer of authenticated objects or information access to other users by changing or increasing the number of owners of that specified resource. However for these same reasons DAC is very labor intensive as each user must define an access for all users to every resource he owns, so its prone to mistakes made by resource owners, and these are very hard to identify as DAC doesn't scale well and is made to work on small systems where the number of users is manageable to not end up with an ACL explosion [15]. DAC offers more flexibility than MAC, however, it provides less security. [14]

2.4.3. Role-Based Access Control (RBAC):

Also known as Rule-Based Access Control, RBAC is the most demanded in regard to access control systems. Not only is it in high demand among households, RBAC has also become highly sought-after in the business world. In RBAC systems, access is assigned by the system administrator and is stringently based on the subject's role within the household or organization and most privileges are based on the limitations defined by their job responsibilities. So, rather than assigning an individual as a security manager, the security manager position already has access control permissions assigned to it. RBAC makes life much easier because rather than assigning multiple individuals particular access, the system administrator only has to assign access to specific job titles. [14]

2.4.4. Attribute-Based Access control (ABAC):

ABAC uses a different approach to access control by using a set of attribute of users, data, environment and a list of policies that allows a high granularity level.

Policies are a set of rules that ABAC uses to grant or deny access to a user to that resource and they are made up three type of attribute , Users to define who or what can access the resource and it can be identity, role , level of the user. Data which determines to what the policy is going

to control and protect and Policy also use the environment attribute (time, place, state etc.) to include the concept of context to the policy.

The weakness of ABAC is its complexity compared to other types of access control and is usually hard to adapt to but it compensates that by being able to accommodate to every type of control and its complexity can be negated by how well it can be implemented and simplified. **Source spécifiée non valide.** [14]

2.4.5. Organization-Based Access Control (OrBAC):

ORBAC is an access control model first presented in 2003. The current approaches of the access control rest on the three entities (subject, action, object) to control the access the policy specifies that some subject has the permission to realize some action on some object. [16]

OrBAC allows the policy designer to define a security policy independently of the implementation. The chosen method to fulfill this goal is the introduction of an abstract level.

- Subjects are abstracted into roles. A role is a set of subjects to which the same security rule apply.
- Similarly, an activity is a set of actions to which the same security rule apply.
- A view is a set of objects to which the same security rule apply. [16]

Each security policy is defined for and by an organization. Thus, the specification of the security policy is completely parameterized by the organization so that it is possible to handle simultaneously several security policies associated with different organizations. The model is not restricted to permissions, but also includes the possibility to specify prohibitions and obligations. From the three abstract entities (roles, activities, views), abstract privileges are defined. And from these abstract privileges, concrete privileges are derived. [16]

OrBAC is context sensitive, so the policy could be expressed dynamically. Furthermore, OrBAC owns concepts of hierarchy (organization, role, activity, view, context) and separation constraints. [16]

2.4.6. Benefits of Access Control

2.4.6.1. Knowing Who's Coming and Going at All Times

Many businesses have equipment and physical assets that are valuable on-site. An access control system keeps track of who's coming and going to ensure that someone hasn't snuck into the building. If a business is large with a lot of employees, it can be difficult for everyone to know who is an employee and who is not. An Access Control System helps prevent strangers from slipping in undetected. [17]

2.4.6.2. Keep Track of Employees

If a business has multiple shifts with large groups of employees coming and going at odd hours, an Access Control System can help organize the chaos and inform you if an employee is in the building when they shouldn't be. It can also help you keep track of who has shown up for work and who hasn't. [17]

2.4.6.3. Secure Sensitive Documents and Data

Many businesses have documents or data that should not be accessible to everyone in the company. An access control system allows a business to limit the access to certain areas that hold hardware or software that this information is saved on. [17]

2.4.6.4.Reduce Theft and Accidents

An Access Control System allows a business to give only approved or specially trained employees access to areas that may hold valuable or dangerous equipment. [17]

2.4.6.5. Multi-Property Protection

An integrated access control system will allow a business to grant access to employees who need to enter multiple or all buildings. [17]

2.4.6.6. No More Worrying About Keys

When an employee quits and fails to return their keys, the business is stuck with the expense of making new keys and possibly even changing the locks. The same would apply when an employee loses his or her company keys. If the employee left on bad terms, this also removes the chance that they will try to re-enter the building [17]

2.4.7. Choosing the Best Access Control System for Your Organization

As you can see, when it comes to choosing the type of access control system that is most suitable for your organization, there are a number of factors involved. Some of those factors include the nature of your business, security procedures within the organization, and the number of users on the system.

Places of business with small or basic applications will probably find Discretionary Access Control to be less complicated and better utilized. If, however, you have highly confidential or sensitive information on your business platform, a Managed Access or Role-Based Access Control system are two options you may want to consider. [17]

2.4.8. Knowing When to Use RBAC vs ABAC

Now that we better understand the major differences between the two models, we can explore best practices for which to use and when. While RBAC and ABAC can be very complex subjects, here are four simple concepts you can refer to not only as you start your IAM implementation, but on an ongoing basis as your organization and needs change:

RBAC is for coarse-grain access control and ABAC is for fine-grain access controls.

When you can make access control decisions with broad strokes, use RBAC. For example, giving all teachers access to Google or all contractors access to email. When you need more granularity than this or need to make a decisions under certain conditions, use ABAC. For example, giving teachers access to Google if they are at School X and teach Grade Y.

Less is More.

If you are creating a lot of very complex RBAC and/or ABAC filters, you are probably doing something wrong. A little bit of planning in advance can help you structure your directory data

in a way that mitigates the need to develop complex filters/queries. However, every now and then, you will definitely have to get creative to establish the right level of access control, but this should be the exception and not the rule.

Divide and Conquer.

You can always use RBAC and ABAC together in a hierarchal approach. For example, using RBAC to control who can see what modules and then using ABAC to control access to what they see (or can do) inside of a module. This is similar to a WAN and LAN-based firewalls where the WAN does the coarse-grain filtering and then LAN-based does the finer-grain inspections.

Just remember, during your implementation of IAM tools, access control is a set of policies that ensures users have the correct access to the correct systems, resources, and applications. So, whether you use RBAC or ABAC, a good IAM solution should help you define what users can do with applications by providing multiple mechanisms to ensure the right people, get the right access, to the right things—at the right time.

2.4.9. Attribute-based Access Control (ABAC):

Definition

Attribute Based Access Control, otherwise known as Policy Based Access Control (PBAC) is typically used to safeguard data in applications, databases, micro services and APIs, within complex architecture.

ABAC is used to grant access based on a user's location, role, the time of day, the device being used, the resource in question, and the desired action, i.e., all the attributes necessary to enforce secure authorization dynamically and in real-time. Put simply, attributes act like the levers in a lock, in that all of them must be aligned – with a policy – before access to data is granted.

Examples of Attribute-Based Access Control

Through ABAC, you can control what end-users can do at both broad and granular levels. You can designate whether the user is an administrator, a specialist user, or an end-user, and align Attributes and access permissions with your employees' positions in the organization. Permissions are allocated only with enough access as needed for employees to do their jobs. [14]

Some of the designations in an ABAC tool can include:

- Management Attribute scope: it limits what objects the Attribute group is allowed to manage.
- Management Attribute group: you can add and remove members.
- Management Attribute: these are the types of tasks that can be performed by a specific role group.
- Management Attribute assignment: this links an attribute to an attribute group.

By adding a user to an attribute group, the user has access to all the attributes in that group. If they are removed, access becomes restricted. Users may also be assigned to multiple groups in the event they need temporary access to certain data or programs and then removed once the project is complete. [14]

Other options for user access may include:

- Primary – the primary contact for a specific account or attribute.
- Billing – access for one end-user to the billing account.
- Technical – assigned to users that perform technical tasks.
- Administrative – access for users that perform administrative tasks.

Benefits of ABAC

Managing and auditing network access is essential to information security. Access can and should be granted on a need-to-know basis. With hundreds or thousands of employees, security is more easily maintained by limiting unnecessary access to sensitive information based on each user's established attribute within the organization. Other advantages include:

Reducing administrative work and IT support

With ABAC, you can reduce the need for paperwork and password changes when an employee is hired or changes their attribute. Instead, you can use ABAC to add and switch attributes quickly and implement them globally across operating systems, platforms and applications. It also reduces the potential for error when assigning user permissions. This reduction in time spent on administrative tasks is just one of several economic benefits of ABAC. ABAC also helps to more easily integrate third-party users into your network by giving them pre-defined attributes. [18]

The Power of Attributes

At the core of ABAC technology is Dynamic Authorization, dynamic authorization is a technology in which authorization and access rights to an organization's network, applications, data, or other sensitive assets are granted dynamically in real-time using attribute-based rules and policies.

The attributes used to define these rules and policies can range from attributes based on the subject, environment, or even the resources that are being accessed. Once the specific set of rules and requirements are met, the specific data will be accessible to the user.

Under ABAC, access decisions can change between requests by simply changing attribute values, without the need to change the subject/object relationships defining underlying rule sets. [19]

Maximizing operational efficiency

ABAC offers a streamlined approach that is logical in definition. Instead of trying to administer lower-level access control, all the attributes can be aligned with the organizational structure of the business and users can do their jobs more efficiently and autonomously. [14]

Improving compliance

All organizations are subject to federal, state and local regulations. With an ABAC system in place, companies can more easily meet statutory and regulatory requirements for privacy and confidentiality as IT departments and executives have the ability to manage how data is being accessed and used. This is especially significant for health care and financial institutions, which manage lots of sensitive data such as PHI and PCI data.

Best practices for implementing ABAC

Implementing a ABAC into your organization shouldn't happen without a great deal of consideration. There are a series of broad steps to bring the team onboard without causing unnecessary confusion and possible workplace irritations. Here are a few things to map out first. [14]

Current Status:

Create a list of every software, hardware and app that has some sort of security. For most of these things, it will be a password. However, you may also want to list server rooms that are under lock and key. Physical security can be a vital part of data protection. Also, list the status of who has access to all of these programs and areas. This will give you a snapshot of your current data scenario.

Current attributes:

Even if you do not have a formal roster and list of attributes, determining what each individual team member does may only take a little discussion. Try to organize the team in such a way that it does not hold back creativity and the current culture.

Write a Policy:

Any changes made need to be written for all current and future employees to see. Even with the use of a ABAC tool, a document clearly articulating your new system will help avoid potential issues.

Make Changes:

Once the current security status and attributes are understood (not to mention a policy is written), it's time to make the changes.

Continually Adapt:

It is likely that the first iteration of ABAC will require some tweaking. Early on, you should evaluate your attributes and security status frequently. Assess first, how well the creative/production process is working and secondly, how secure your process happens to be.

A core business function of any organization is protecting data. An ABAC system can ensure the company's information meets privacy and confidentiality regulations. Furthermore, it can secure key business processes, including access to IP, that affect the business from a competitive standpoint.

Chapter 3: Conception and realization

3.1. Implementation

We will present in this chapter the realization part of our application which aims to implement the solution described in the previous chapters. To do this, we will first start by describing the proposed approach. Next, we describe the implementation of the approaches. Finally, we will present the main graphical interfaces of our software.

In most cases the ABE systems are based on the use of a set of attributes defined under an access structure associated with an encryption algorithm, we can use both CP ABE or KP ABE.

In contradiction to that our proposition aims to better the CP ABE and to better the Confidentiality, availability and integrity of data in a CP ABE system, we proposed to compress the access structure into a QR code that will represent the digital signature of a user based on the proposed attributes.

3.2. QR Authentication

In order to implement a QR authentication system to our software we will use the API ZXing (“Zebra Crossing”) for QR code processing in Java. Its library has multiple components and we will be using the ‘core’ for QR code creation in our system.

To realize our authentication system we used two functions :

The function **generateQRCodeImage** takes a string in our case (the id of the user that just registered and his attributes) and as an output give a QR that the user will use for his authentication.

```
private static final String QR_CODE_IMAGE_PATH = "./MyQRCode.png";

public static void generateQRCodeImage(String text, int width, int height, String filePath)
    throws WriterException, IOException {
    QRCodeWriter qrCodeWriter = new QRCodeWriter();
    BitMatrix bitMatrix = qrCodeWriter.encode(text, BarcodeFormat.QR_CODE, width, height);

    Path path = FileSystems.getDefault().getPath(filePath);
    MatrixToImageWriter.writeToPath(bitMatrix, "PNG", path);
}
```

Figure 18:generate QR code image

and `readQRCode` takes a QR code and extract the user id in order to log him in

```
public static String readQRCode(String fileName) {
    File file = new File(fileName);
    BufferedImage image = null;
    BinaryBitmap bitmap = null;
    Result result = null;

    try {
        image = ImageIO.read(file);
        int[] pixels = image.getRGB(0, 0, image.getWidth(), image.getHeight(), null, 0, image.getWidth());
        RGBLuminanceSource source = new RGBLuminanceSource(image.getWidth(), image.getHeight(), pixels);
        bitmap = new BinaryBitmap(new HybridBinarizer(source));
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    if (bitmap == null)
        return null;

    QRCodeReader reader = new QRCodeReader();
    try {
        result = reader.decode(bitmap);
        return result.getText();
    } catch (NotFoundException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (ChecksumException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (FormatException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    return null;
}
```

Figure 19: read QR code image

3.3. Encryption and Access control

For the implementation of the cp abe we constructed the keys in sort so that the keys are a series of 7 bits each representing an attribute of the seven attributes proposed 0 represent that the following attribute isn't met while 1 represent that the attribute is satisfied.

The last is ensured by using the function **GenerateKey**, that takes as argument a string representing a set of argument.

```
private void generateKey(String attributes) {

    String[] attri = attributes.split(" ");
    int j = 0 ;

    for (int i = 0; i < attri.length; i++) {

        System.out.println("attri is = " +attri[i]);
        if (attri[i].equals("/UIDESIGNER") || attri[i].equals("/UXDESIGNER") || attri[i].equals("/FRONTDEV"))

            if (attri[i].equals("/UIDESIGNER")) {
                attri[i] = ""+1;
            }
            if (attri[i].equals("/UXDESIGNER")) {
                attri[i] = ""+10;
            }
            if (attri[i].equals("/FRONTDEV")) {
                attri[i] = ""+100;
            }
            if (attri[i].equals("/BACKDEV")) {
                attri[i] = ""+1000;
            }
            if (attri[i].equals("/FULLDEV")) {
                attri[i] = ""+10000;
            }
            if (attri[i].equals("/QUALITYENGINEER")) {
                attri[i] = ""+100000;
            }
            if (attri[i].equals("/TESTER")) {
                attri[i] = ""+1000000;
            }
        } else {
            attri[i] = ""+0;
        }
    }

    for (int i = 0; i < attri.length; i++) {
        System.out.println("adding = "+attri[i]);
        j = j + Integer.parseInt(attri[i]);
    }
    keyValue = j ;
    // /UI DESIGNER /UX DESIGNER /FRONT DEV /BACK DEV /FULL DEV /FULL DEV /QUALITY ENGINEER /TESTER

    key = attri ;
}
```

Figure 20: key generation

User's keys can be generated through a Boolean combination of the available users' attributes.

We do have seven (07) attributes so our key would be in the form of "XXXXXXX ", where X =zero (0) or X =one (1)

To encrypt files the system only takes account of the policy set by the data owner and encrypt the file with a key generated from the policy.

Each file having a policy set by the owner of that resource

Once the file is encrypted the function **hasAccess** will take as argument the policy of the file and the key of the user wanting to access the file the function verify if the policy is met:

```
public static boolean HasAccess(String k, String p, Policy po) {
    k = "000000" + k ;
    String [] AccessList = p.split(" ");
    String [] key = k.split("(?!^)");

    key = reverse(key, key.length);

    Boolean t = false ;

    if (AccessList[0].equals("UIDESIGNER")) {
        if (key[0].equals("1")) {
            t = true ;
        }
    }
    if (AccessList[0].equals("UXDESIGNER")) {
        if (key[1].equals("1")) {
            t = true ;
        }
    }
    if (AccessList[0].equals("FRONTDEV")) {
        if (key[2].equals("1")) {
            t = true ;
        }
    }
    if (AccessList[0].equals("BACKDEV")) {
        if (key[3].equals("1")) {
            t = true ;
        }
    }
    if (AccessList[0].equals("FULLDEV")) {
        if (key[4].equals("1")) {
            t = true ;
        }
    }
    if (AccessList[0].equals("QUALITYENGINEER")) {
        if (key[5].equals("1")) {
            t = true ;
        }
    }
    if (AccessList[0].equals("TESTER")) {
        if (key[6].equals("1")) {
            t = true ;
        }
    }
}
```

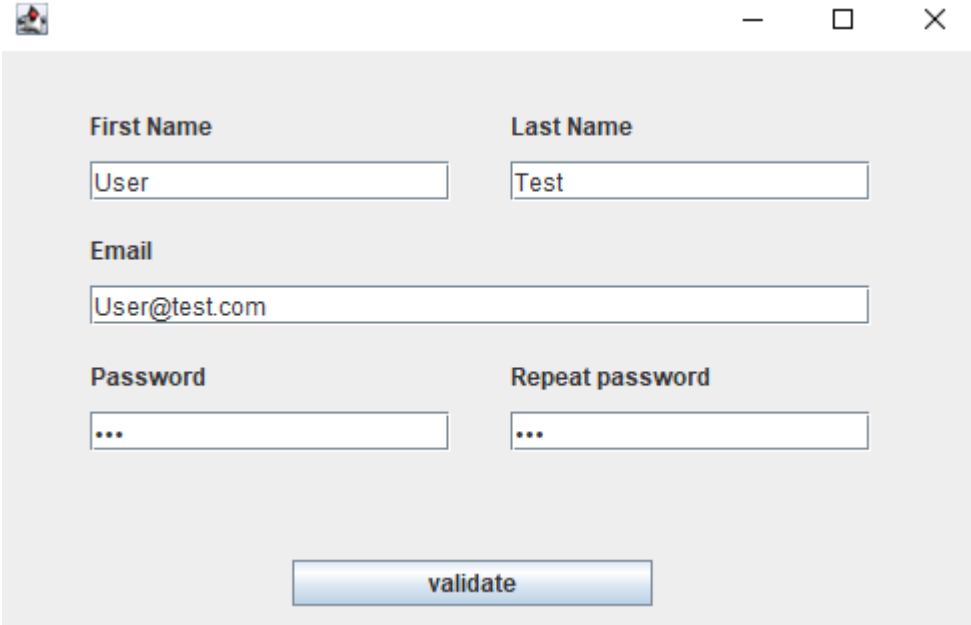
Figure 21: access verification

If the policy is satisfied the user gets access to the private key in order to access the file itself with the function **getMk**.

3.4. Software Presentation

Each user must identify himself before accessing any part of our program, so he would mainly start with the authentication either by signing in or by signing up and he cannot have any access without identifying himself.

To sign up, the user must fill a form with the required information and choose the attributes in order to generate a unique QR code that will be used later to sign in.



First Name: User

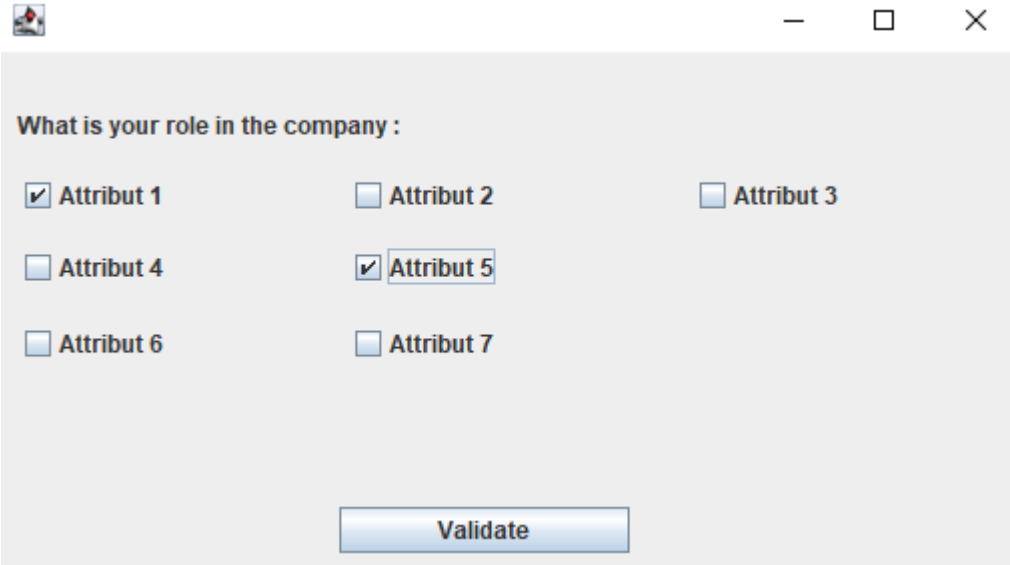
Last Name: Test

Email: User@test.com

Password: ...

Repeat password: ...

validate



What is your role in the company :

Attribut 1 Attribut 2 Attribut 3

Attribut 4 Attribut 5

Attribut 6 Attribut 7

Validate

Figure 22: register frame

To sign in, the user must upload the QR code using a file chooser utility and the system will transform that picture into data and will compare it with the database. If the user already exists,

he will login in and continue using the app. Else, he will see an error popup of a failed authentication and he must choose another QR code or sign up.

3.4.1. Users platforms

After logging in or signing up, users can be generalized under three roles with different levels of access to the resources:

Admin:

Which is the highest level of accessibility the following user dispose of 2 main graphic interfaces one of which he can access other users grant them higher access or revoke privileges

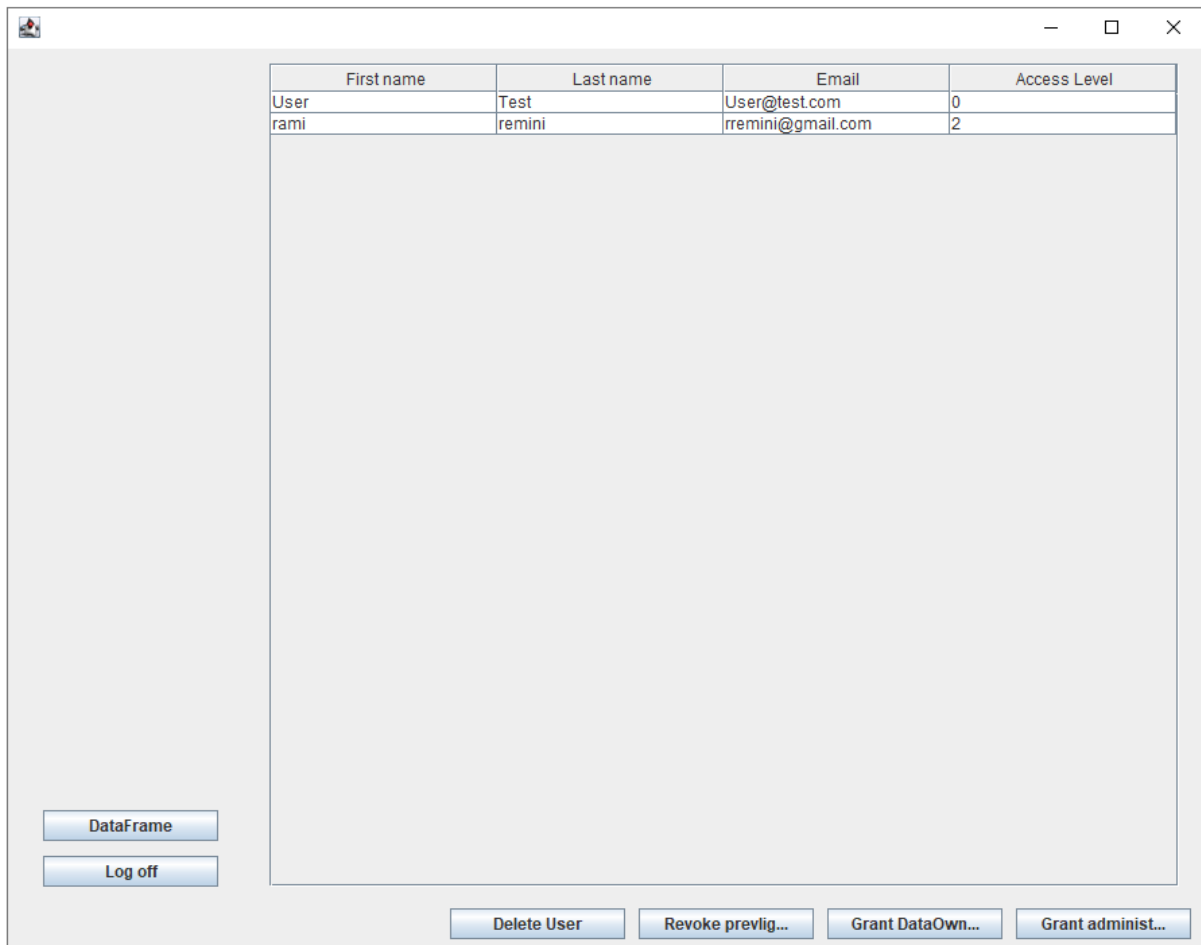


Figure 23:login frame

Data owner

Second to the admin role is the data owner the user with this level of access have the right to write file and publish them in the system although they can only access to their own file

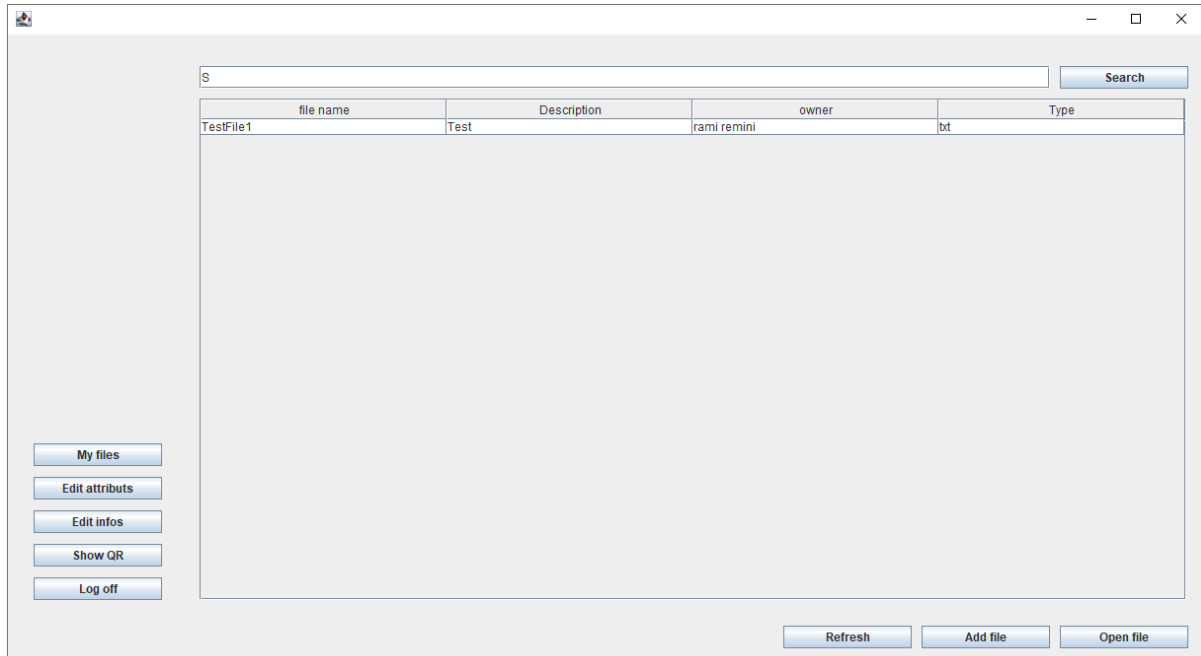
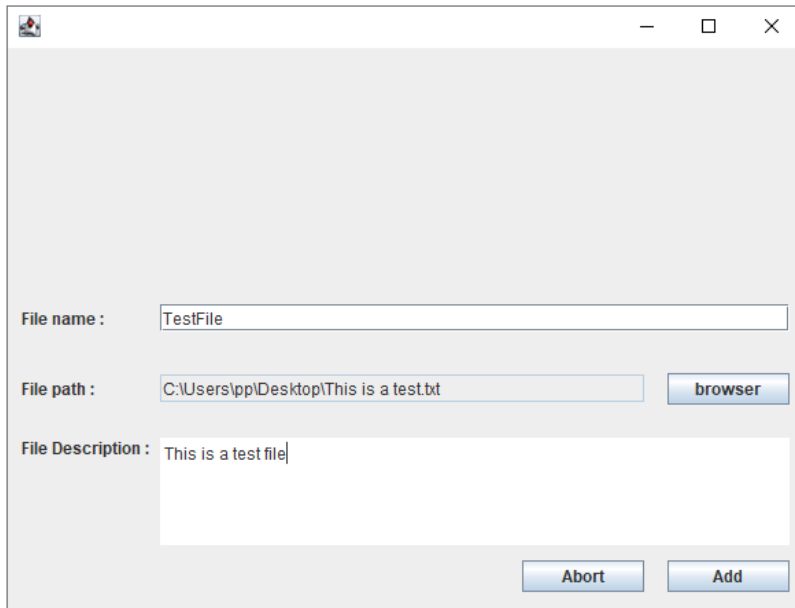


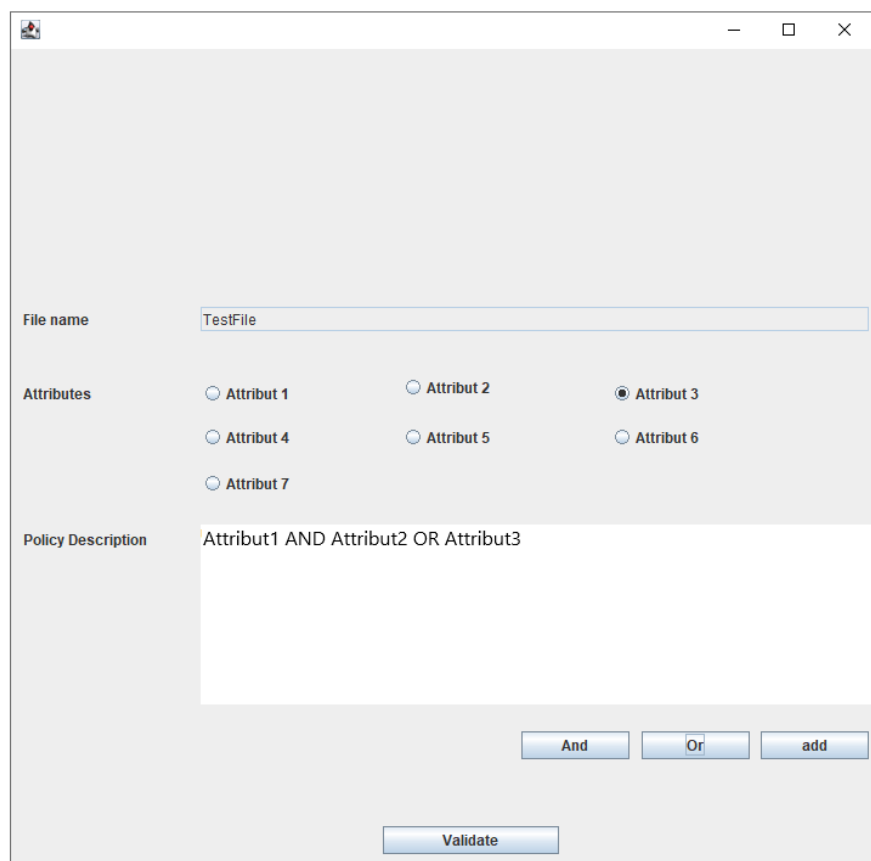
Figure 24: data frame

In order for these users to publish files they need to go through 2 in which the user enter information about the file he is submitting and in the second he select attribute needed to decrypt the file



A screenshot of a software window with a title bar containing a small icon, a minus sign, a maximize button, and a close button. The window contains the following fields and controls:

- File name :** A text input field containing "TestFile".
- File path :** A text input field containing "C:\Users\pp\Desktop\This is a test.txt" and a "browser" button to its right.
- File Description :** A larger text input field containing "This is a test file".
- At the bottom right, there are two buttons: "Abort" and "Add".



A screenshot of a software window with a title bar containing a small icon, a minus sign, a maximize button, and a close button. The window contains the following fields and controls:

- File name** : A text input field containing "TestFile".
- Attributes** : A group of seven radio buttons labeled "Attribut 1" through "Attribut 7". "Attribut 3" is selected.
- Policy Description** : A text input field containing "Attribut1 AND Attribut2 OR Attribut3".
- At the bottom right, there are three buttons: "And", "Or", and "add".
- At the bottom center, there is a "Validate" button.

Figure 25 encryption frame

For a user to decrypt a file his eligibility to access that certain resource is verified in the decryption frame

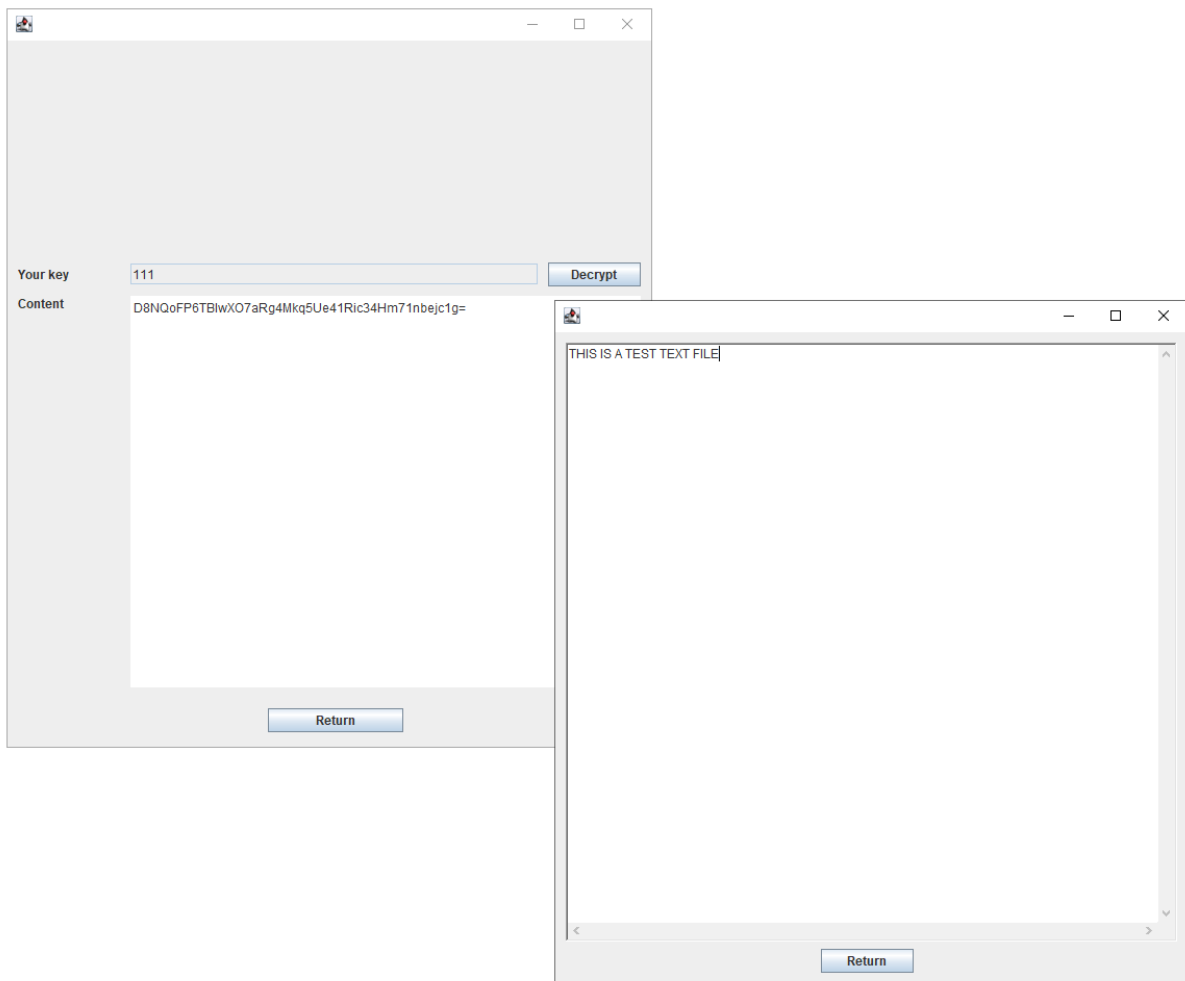


Figure 28:decrypt frame

3.5. Conclusion

In this chapter, we have described the process of making our application, specifying the development tools and libraries used. We presented the graphical interfaces that make up the spaces of the different users of our system while exposing the functions that make up each space. We finished the implementation of our application while respecting the design developed.

GENERAL CONCLUSION

The objective of this work is to find a solution to the problem of data security in order to be able to share them with users authorized while maintaining confidentiality, integrity and privacy.

To achieve our objective, we made a bibliographic study that we distributed

in two chapters:

- In chapter 1, we introduced basic concepts on information systems and the ways to secure them
- In chapter 2, we explored security mechanisms (encryption and access control and different ways of authentication) to add another layer of security to data. We are subject to attribute-based access control and CP-ABE encryption.

Afterwards, we proposed a solution that combines two security approaches:

- A QR Code authentication system
- An RBAC access control with the CP-ABE attribute encryption algorithm which encrypt folders before saving them in the system.

By combining these approaches, we were able to achieve the security objectives required in this domain :

- Access Control: Determine which users are allowed to access the data.
- Privacy: Data in the system is encrypted and only people authorized can access it.
- Traceability: which keeps track of the state and movement of information,

However, several improvements that can be made to our platform, for example:

- Adding other concrete actors to a real scenario.
- adding a two factor authentication.
- the possibility of adding a mechanism that keeps track of the state and movements of information.

Bibliography

- [1] U. D. o. Defense, *The dictionary of Military and Associated Terms*.
- [2] «H. I. Hahn, J. K. Joung,» “Implementation of algorithm to decode two-dimensional barcode PDF-417,” *Int. Conf.*.
- [3] «“Paper-based document security–A Review,” in European Conf. on Security and Detection,,» *Renesse, R.L. van*.
- [4]] Sridevi, C.: ‘A survey on network security’, *Global Journal of Computer Science and Technology*, 2018.
- [5] AL-DAHMAN, Ruqayah R., SHI, Qi, LEE, Gyu Myoung, et al. Survey on revocation in ciphertext-policy attribute-based encryption. *Sensors*, 2019, vol. 19, no 7, p. 1695..
- [6] DEVIGNE, Julien. *Protocoles de re-chiffrement pour le stockage de données*. 2013. Doctorat thesis. Caen University.
- [7] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” *Proc. ACM Conf. Comput. Commun. Secur.*, pp. 89–98, 2006.
- [8] Hadji, F. (2018). *Conception et réalisation d'un système de cryptage pour les images médicales* (Doctorat thesis, Mohamed Boudiaf de M'Sila university)..
- [9] de Mello, Flávio Luis, and Jose AM Xexeo. "Identifying Encryption Algorithms in ECB and CBC Modes Using Computational Intelligence." *J. Univers. Comput. Sci.* 24.1 (2018): 25-42..
- [10] «RSA Cryptography Standard, PKCS #1 v2.1, 2002.».

- [11] Fujdiak, R., Masek, P., Hosek, J., Mlynek, P., and Misurec, J.: 'Efficiency evaluation of different types of cryptography curves on low-power devices', in Editor (Ed.)^(Eds.): 'Book Efficiency evaluation of different types of cryptography curves on low-po.
- [12] FOTOHI, Reza, FIROOZI BARI, Somayyeh, et YUSEFI, Mehdi. Securing wireless sensor networks against denial-of-sleep attacks using RSA cryptography algorithm and interlock protocol. *International Journal of Communication Systems*, 2020, vol. 33, no 4, p. e423.
- [13] JI, Yi-mu, TAN, Jie, LIU, Hai, et al. A Privacy Protection Method Based on CP-ABE and KP-ABE for Cloud Computing. *J. Softw.*, 2014, vol. 9, no 6, p. 1367-1375..
- [14] F. C. & K. by Hu, *Attribute-Based Access Control*, 2017.
- [15] *CAMELOT*, "DIFFERENTIATING BETWEEN ACCESS CONTROL TERMS",.
- [16] *Or-BAC Organization Based Access Control*. A. Miège, and F. Cuppens. *Le Croisic*,.
- [17] *Organization-based access control*. A. Abou El Kalam, P. Balbiani, S. Benferhat, F..
- [18] Yuan, Eric, and Jin Tong. "Attributed based access control (ABAC) for web services." *IEEE International Conference on Web Services (ICWS'05)*. IEEE, 2005..
- [19] Medinilla, Victoria R., et al. "Impact of ammonia-based aeration control (ABAC) on energy consumption." *Applied Sciences* 10.15 : 5227., 2020.
- [20] "Born to be breached" by Sean Gallagher on Nov 3 2012. *Arstechnica*. Retrieved from <http://arstechnica.com/information-technology/2012/11/born-to-be-breached-the-worst-passwords-are-still-the-most-common/> on May 15, 2013.
- [21] N. Döttling and S. Garg, "Identity-based encryption from the diffie-hellman assumption," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10401 LNCS, pp. 537–569, 2017.
- [22] «RSA Cryptography Standard, PKCS #1 v2.1, 2002.».

- [23] «J. Z. Gao, “Understanding 2D-barcode technology and applications in M-commerce – design and implementation».
- [24] «Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm».
- [25] «P. Kuacharoen, “Design and analysis of methods for signing electronic documents using mobile phones,” Int. Conf.».
- [26] «Digital Signature Standard (DSS), FIPS PUB 186-3, 2009.».
- [27] «Gao, J. Z.,» *“Understanding 2D-barcode technology and applications in M-commerce – design and implementation.”*
- [28] *SANS Institute. "A Short Primer for Developing Security Policies." Accessed from http://www.sans.org/security-resources/policies/Policy_Primer.pdf on May 31, 2013.*
- [29] «www.sans.org/score/checklists/mobile-device-checklist.xls, Taken from SANS Institute's Mobile Device Checklist. You can review the full checklist at,» [En ligne].
- [30] *http://www.fas-it.fas.harvard.edu/services/student/policies/rules_and_responsibilities.*
- [31] *<https://bus206.pressbooks.com/chapter/chapter-6-information-systems-security/>.*
- [32] «Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm».
- [33] *<https://www.midwestalarmservices.com/our-company/blog/3-types-access-control-which-right-your-building>.*
- [34] «R.L. van Renesse, “Paper-based document security–A Review,” in European Conf. on Security and Detection,».

- [35] «WARASART, Maykin et KUACHAROEN, Pramote. Based document authentication using digital signature and QR code. In : 4TH International Conference on Computer Engineering and Technology (ICCET 2012). 2012.».
- [36] «“Design and analysis of methods for signing electronic documents using mobile phones,” Int. Conf.» *Kuacharoen, P.*