وزارة التعليم العالي والبحث العلمي

جامعة سعيدة. مولاي الطاهر

كلية العلوم

قسم: الإعلام الآلي

Mémoire de Master

Spécialité : Réseaux Informatique et Systèmes Répartis

Thème

Un Modèle De Chiffrement Pour Préserver La Confidentialité Dans Les Clouds

Présenté par :

- Daoudi Mohamed Amine
- Semir Fatima

Dirigé par :

- Mr : Benyahia kadda

Promotion 2023 - 2024

Remerciements

*N*ous tenons à exprimer notre profonde gratitude à tous ceux qui ont contribué à la réalisation de cette recherche, à commencer par Dieu qui nous a donné la force et la patience, ainsi que notre cher superviseur Mr**.** ***Benyahia kadda*** qui nous a fourni des documents intéressants et des conseils précieux, et qui nous a apporté un soutien continu. Nous le remercions pour son attitude noble et l'impact important qu'il a eu sur nous.

Nous tenons également à exprimer notre gratitude aux membres du comité d'évaluation, Mr ***Zarouki kadda*** et M^{me} ***Dib Soumia*** pour avoir accepté d'examiner et d'évaluer notre travail.

*N*ous remercions également tous les professeurs et enseignants qui nous ont transmis leur savoir et nous ont accueillis dans leurs salles de classe, faisant preuve de patience et des compétences pédagogiques nécessaires pour nous former et répondre à nos questions.

*E*nfin, nous tenons à exprimer notre sincère gratitude à nos chers amis qui nous ont aidés et encouragés à mener à bien cette recherche.

Dédicace

Je tiens à exprimer ma profonde gratitude à tous ceux qui m'ont aidé dans mon parcours scolaire, et je dédie ce fruit à leur précieuse contribution.

Je remercie Allah le Miséricordieux, et je dédie ce fruit à tous ceux qui m'ont soutenu et ont été à mes côtés dans mon voyage scolaire.

Je le dédie à ma chère mère et à mon cher père, qui m'ont offert affection et soutien tout au long de ma vie, et à une âme pure qui réside dans mon cœur.

Je le dédie à ma chère sœur et à mes chers frères, ainsi qu'à mon fils bien-aimé et à la prunelle de mes yeux, "Wassim Abdelnour djalloul".

Je le dédie à toute la famille "Daoudi", à mes fidèles amis, et à tous mes amis de la promotion 2024.

Amine

Dédicace

*U*ne chance m'a été offerte aujourd'hui pour parler Des personnes qui me sont très chères.

*A*vant tout; je remercie Allah» d'avoir donné Le courage et la patience pour réaliser ce travail. A mes chers parents Mon Pere et Ma Mère Qui ont contribué a ma réussite et qui m'ont ande A devenir ce que je suis, en reconnaissance de leur Confiance et leur sacrifice.

*M*es frères. A toute ma famille Semir A tous les étudiants de 2 année maste RISR

*P*romotion 2024. A tous que j'aime et je respect.

Fatima..

Résumé

Les informations circulent à travers le réseau via des chemins différents, et il est nécessaire de les sécuriser pour contrôler l'accès non autorisé aux données sous toutes leurs formes. La sécurité des données est assurée en convertissant le texte normal en une forme illisible à l'aide d'algorithmes de cryptage. Les techniques de cryptage reposent sur le chiffrement des messages pour garantir une communication sécurisée à travers le réseau. Il est essentiel pour les entreprises et les particuliers de sécuriser leurs informations contre les attaques et les pirates informatiques afin de garantir la confidentialité et l'intégrité des données. Pour renforcer la sécurité des données sur le réseau, il est nécessaire de crypter les données de manière illisible. De nombreuses recherches ont été menées sur l'évolution des systèmes de cryptage, cependant, l'évolution actuelle dans ce domaine est le cryptage de l'acide ribonucléique qui a été développé en fonction de la capacité de calcul de l'ADN. La recherche propose une méthode pour exécuter la dissimulation des données en utilisant des chaînes d'ADN. Les concepts de codage binaire et de valeurs aléatoires sont utilisés, et une clé est créée pour crypter les données. L'expéditeur partage la chaîne d'ADN représentant les données cryptées extraites des données d'origine. De cette manière, il est assuré que le traitement des informations dans le Cloud Computing est effectué sans identifier les données d'origine. De plus, nous avons évalué l'algorithme de cryptage en tenant compte de nombreux facteurs, y compris la longueur de la clé, les performances et la consommation de ressources, afin de prendre la décision appropriée.

Mots clés *: Cloud Computing, Les algorithmes de cryptage, la confidentialité et l'intégrité des données, la chaîne d'ADN*

Abstract

Information circulates through the network via different paths, and it is necessary to secure it to control unauthorized access to data in all its forms. Data security is ensured by converting plain text into an unreadable form using encryption algorithms. Encryption techniques rely on encrypting messages to ensure secure communication across the network. It is essential for businesses and individuals to secure their information against attacks and hackers in order to guarantee data confidentiality and integrity. To enhance data security on the network, it is necessary to encrypt data in an unreadable manner. Many researches have been conducted on the evolution of encryption systems, however, the current evolution in this field is RNA encryption, which has been developed based on DNA's computing capacity. The research proposes a method for data concealment using DNA strands. Binary coding and random values concepts are used, and a key is created to encrypt the data. The sender shares the DNA strand representing the encrypted data extracted from the original data. This ensures that the processing of information in Cloud Computing is done without identifying the original data. Furthermore, we evaluated the encryption algorithm considering various factors, including key length, performance, and resource consumption, in order to make the appropriate decision.

***Keywords**: Cloud Computing, encryption algorithms, data confidentiality and integrity, DNA strand.*

الملخص

تنتقل المعلومات عبر الشبكة عبر مسارات مختلفة، ومن الضروري تأمينها للسيطرة على الوصول غير المصرح به إلى البيانات بجميع أشكالها. يتم ضمان أمان البيانات عن طريق تحويل النص العادي إلى شكل غير قابل للقراءة باستخدام خوارزميات التشفير. تعتمد تقنيات التشفير على تشفير الرسائل لضمان اتصال آمن عبر الشبكة. من الضروري للشركات والأفراد تأمين معلوماتهم ضد الهجمات وقراصنة الحواسيب لضمان سرية وسلامة البيانات. لتعزيز أمان البيانات على الشبكة، من الضروري تشفير البيانات بشكل غير قابل للقراءة. تم إجراء العديد من الأبحاث حول تطور أنظمة التشفير، ومع ذلك، التطور الحالي في هذا المجال هو تشفير الحمض النووي الريبوزي الذي تم تطويره بناءً على قدرة حساب الحمض النووي. تقترح البحث طريقة لتنفيذ إخفاء البيانات باستخدام سلاسل الحمض النووي. يتم استخدام مفاهيم الترميز الثنائي والقيم العشوائية، ويتم إنشاء مفتاح لتشفير البيانات. يشارك المرسل سلسلة الحمض النووي التي تمثل البيانات المشفرة المستخرجة من البيانات الأصلية. بهذه الطريقة، يتم ضمان معالجة المعلومات في الحوسبة السحابية دون تحديد البيانات الأصلية. بالإضافة إلى ذلك، قمنا بتقييم خوارزمية التشفير مع مراعاة العديد من العوامل، بما في ذلك طول المفتاح والأداء واستهلاك الموارد، لاتخاذ القرار المناسب.

الكلمات المفتاحية: الحوسبة السحابية،خوارزميات التشفير، سرية وسلامة البيانات، سلسلة الحمض النووي

TABLE DES MATIÈRES

1 Présentation Générale des Concepts Cloud Computing

2 La sécurité et la confidentialité dans le Cloud

5 Implémentation et Évaluation

LISTE DES TABLEAUX

LISTE DES FIGURES

Introduction Générale

Dans notre ère numérique, la valeur et l'importance des données augmentent. Le stockage et le transfert de données sensibles en ligne nécessitent des niveaux élevés de sécurité pour protéger les informations contre tout accès non autorisé ou utilisation malveillante. Le chiffrement joue un rôle vital dans la réalisation de cette sécurité.

Le chiffrement fait référence au processus de conversion des données en une forme illisible pour empêcher tout accès non autorisé. Des algorithmes complexes sont utilisés pour convertir les données d'origine (textes, images, vidéos, etc.) en un "texte chiffré" qui ne peut être compris qu'en utilisant la clé de déchiffrement correcte.

Les méthodes traditionnelles de chiffrement, telles que le chiffrement AES, sont efficaces pour protéger les données contre les menaces actuelles. Cependant, avec l'émergence de puissants ordinateurs quantiques, la possibilité de compromettre ces méthodes augmente.

C'est là que le chiffrement a base d'ADN comme solution prometteuse pour faire face aux défis de sécurité à l'ère quantique intervient. Le chiffrement utilisant l'ADN est une technologie émergente et prometteuse pour protéger les données dans le Cloud Computing.

Cette technologie offre des avantages uniques tels que la résistance quantique, la stabilité à long terme et l'efficacité de stockage. On s'attend à ce qu'elle joue un rôle important dans la garantie de la sécurité des données à l'ère quantique alors qu'elle continue à évoluer.

Chapitre 01

Présentation Générale des Concepts Cloud Computing

1 Introduction :

Le Cloud Computing est un concept extrêmement puissant dans cette ère de l'évolution technologique émergente, utilisé à la fois pour les applications grand public et professionnelles. Il s'agit d'un mécanisme de calcul centralisé vers lequel de nombreuses organisations se sont tournées au cours de la dernière décennie. Dans cette section, nous allons présenter les concepts du Cloud Computing ainsi que son architecture.

Dans ce chapitre nous allons présenter les notions fondamentales du Cloud Computing, ses enjeux, ses évolutions et son utilité ainsi que la technologie qui la constitue et les différents acteurs du secteur. Nous devons dans un premier temps étudier le Cloud Computing de manière générale (Définitions, Avantages, inconvénients…), dans un second temps nous allons étudier les trois services principaux, sur lesquels le Cloud Computing repose: applicatif, plateforme, infrastructure, qui ont donné naissance aux fameux **SaaS/PaaS/IaaS**. Et la dernière partie de ce chapitre présente les différents avantages et inconvénient du Cloud Computing, et met l'accent sur l'aspect de la sécurité du Cloud Computing.

2 Historique

Techniquement, le concept de Cloud Computing est loin d'être nouveau, il est même présent depuis des décennies. On en trouve les premières traces dans les années 1960, quand John McCarthy[1] affirmait que cette puissance de traitement informatique serait accessible au public dans le futur. Le terme en lui-même est apparu plus couramment aux alentours de la fin du XXe siècle et il semblerait qu'Amazon.com soit l'un des premiers à avoir assemblé des data-center et fournit des accès à des clients. Les entreprises comme IBM et Google ainsi que plusieurs universités ont seulement commencé à s'y intéresser sérieusement aux alentours de 2008, quand le Cloud Computing est devenu un concept à la mode. Réalisant ce qu'ils pourraient faire de toute cette puissance, de nombreuses compagnies ont ensuite commencé à montrer un certain intérêt, puis à échanger leurs anciennes infrastructures et applications internes contre ce que l'on appelle : les pay per-use service (services payés à l'utilisation). [02] C'est la cinquième génération de l'informatique après les Mainframes, les PCs, les Clients/Serveurs et le Web. [01]

Auparavant, seuls les superordinateurs étaient capables de fournir une telle puissance et étaient principalement utilisés par les gouvernements, les militaires, les laboratoires et

[1]John McCarthy *(né le 4 septembre 1927, à Boston, Massachusetts)* est le principal pionnier de l'intelligence artificielle .Il est également l'inventeur en **1958** du langage Lisp. A la fin des années **1950**, il a créé avec Fernando Cobarto la technique du temps partagé, qui permet à plusieurs utilisateurs d'employer simultanément un même ordinateur

les universités pour effectuer des calculs complexes tels que la prédiction du comportement d'un avion en vol, les changements climatiques ou la simulation d'explosions nucléaires. Aujourd'hui, des entreprises comme Google proposent des applications qui exploitent cette même puissance et sont accessibles à tout moment, de n'importe où et par n'importe qui via Internet. Certaines universités prestigieuses ont également lancé leurs propres programmes de Cloud Computing en offrant l'accès à des réseaux de centaines voire de milliers de processeurs. Des entreprises comme IBM ont récemment annoncé leur intention d'utiliser massivement le Cloud Computing à l'avenir. Ils ont récemment dévoilé un système ultra-performant appelé "Blue Cloud" qui permettra aux banques et aux entreprises diverses de distribuer leurs calculs sur un grand nombre de machines sans avoir besoin d'une infrastructure interne.

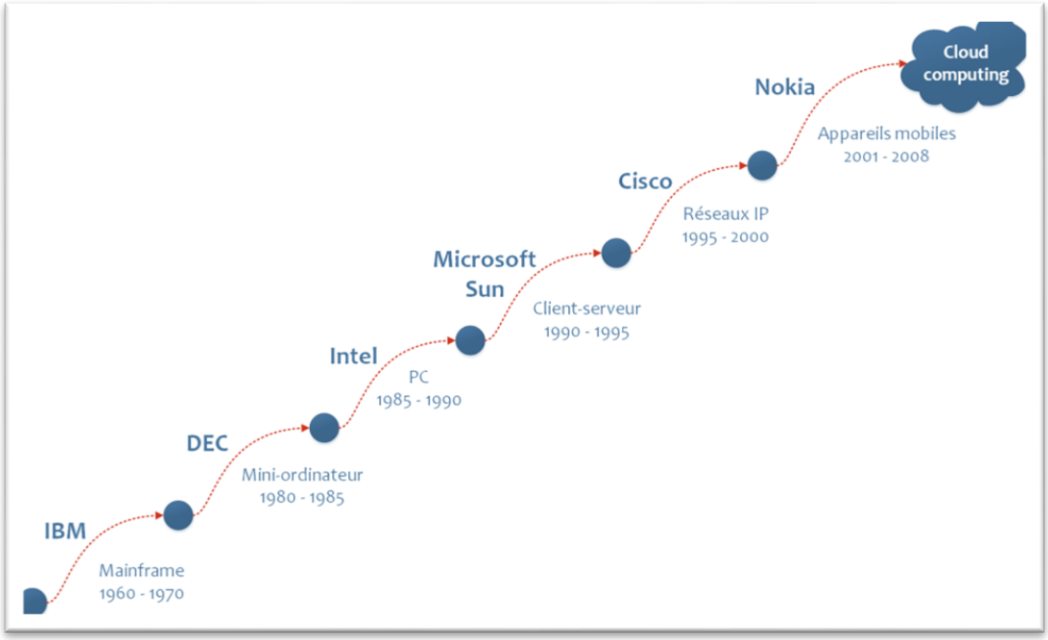


Figure I.1: *Evolution de l'informatique* [01]

3 Définition du Cloud :

Il existe de nombreuses définitions du de cloud computing, et la définition la plus répandue est celle du NIST.

NIST définit le cloud comme étant *« Le Cloud Computing est un modèle qui permet un accès réseau pratique et sur demande à un pool partagé de ressources informatiques configurables (par exemple, des réseaux, des serveurs, du stockage, des applications et des services) qui peut être rapidement approvisionné et disponible sans trop d'efforts de gestion ou d'interaction d'opérateurs » (NIST)*

En d'autres termes, Le Cloud Computing, littéralement l'informatique dans les nuages est un concept qui consiste à déporter sur des serveurs distants des stockages et des traitements informatiques traditionnellement localisés sur des serveurs locaux ou sur le poste de l'utilisateur. Il consiste à proposer des services informatiques sous forme de service à la demande, accessible de n'importe où, n'importe quand et par n'importe qui, grâce à un système d'identification, via un PC et une connexion à Internet. Cette définition est loin d'être simple à comprendre, toutefois l'idée principale à retenir est que le Cloud n'est pas un ensemble de technologies, mais un modèle de fourniture, de gestion et de consommation de services et de ressources informatiques [3]. Pour Wikipédia, il s'agit *:«d'un concept de déportation sur des serveurs distants des traitements informatiques traditionnellement localisés sur le poste client »*.

Pour CISCO *: « Le Cloud Computing est une plateforme de mutualisation informatique fournissant aux entreprises des services à la demande avec l'illusion d'une infinité de ressources »* [4].

4 But du Cloud Computing

L'environnement de Cloud Computing vise à plusieurs buts, nous pouvons citer par exemple:

Fournir une infrastructure, plate-forme et application comme des services payants dans un modèle « *pay-as-You-go* » aux consommateurs.

Faire fonctionner les Datacenter par leur structuration comme un réseau de services virtuels (*matériel, base de données, interface utilisateur*) pour que les utilisateurs soient capables d'avoir accès et déployer des applications partout dans le monde à la demande aux coûts compétitifs selon leurs exigences en QoS.

Libérer les sociétés d'information de la tâche d'installation basique du matériel *(des serveurs)* et des infrastructures logicielles et permettre ainsi plus la focalisation sur l'innovation et la création des valeurs d'entreprises.

5 Caractéristiques du Cloud Computing :

Les services de Cloud Computing (NIST) sont définis par cinq caractéristiques principales :

- ***Libre-service à la demande :*** Les utilisateurs peuvent allouer et libérer des ressources à distance en temps réel, sans intervention de l'opérateur.
- ***Large accès au réseau :*** Toutes les ressources partagées doivent être accessibles et disponibles universellement et facilement via le réseau, quel que soit le type de client *(serveur, PC, client mobile, etc.).*

- ***Mise en commun des ressources :*** Les ressources telles que la bande passante réseau, les machines virtuelles, la mémoire, la puissance de traitement et la capacité de stockage sont partagées pour servir plusieurs clients selon un modèle multi-locataire. Les ressources virtuelles et physiques sont allouées et réallouées dynamiquement en fonction des besoins et des clients.

- ***Élasticité rapide :*** Les ressources et les capacités peuvent être déployées et mises à l'échelle rapidement et automatiquement, en quantité et à tout moment selon la demande.

- ***Service mesuré :*** Les systèmes Cloud doivent être capables de s'autocontrôler et de se gérer pour optimiser leur fonctionnement interne. Cela est réalisé grâce à des mesures de référence obtenues par divers mécanismes de supervision.

Un service ne peut pas être considéré comme un service de Cloud Computing s'il ne satisfait pas ces caractéristiques.

6 Eléments constitutifs du Cloud Computing :

Les éléments pouvant constitué le système Cloud sont les suivants :

6.1 ***La virtualisation*** :La virtualisation se définit comme l'ensemble des techniques matérielles et/ou logiciels qui permettent de faire fonctionner sur une seule machine, plusieurs systèmes d'exploitation *(appelées machines virtuelles (VM), ou encore OS invitée)* [6]. La virtualisation des serveurs permet une plus grande modularité dans la répartition des charges et la reconfiguration des serveurs en cas d'évolution ou de défaillance momentanée

6.2 ***Le Datacenter :***

Un centre de traitement de données *(data center en anglais)* est un site physique sur lequel se trouvent regroupés des équipements constituants le système d'information de l'entreprise *(mainframes, serveurs, baies de stockage, équipements réseaux et de télécommunications, etc.)*. Il peut être interne et/ou externe à l'entreprise, exploité ou non avec le soutien des prestataires. Il comprend en général un contrôle sur l'environnement *(climatisation, système de prévention contre l'incendie, etc.),* une alimentation d'urgence et redondante, ainsi qu'une sécurité physique élevée.

Cette infrastructure peut être propre à une entreprise et utilisé par elle seule ou à des fins commerciaux. Ainsi, des particuliers ou des entreprises peuvent venir y stocker leurs données suivant des modalités bien définies [7].

6.3 La Plateforme collaborative :

Une plate-forme de travail collaboratif est un espace de travail virtuel. C'est un site qui centralise tous les outils liés à la conduite d'un projet et les met à disposition des acteurs.

L'objectif du travail collaboratif est de faciliter et d'optimiser la communication entre les individus dans le cadre du travail ou d'une tâche.

Les plates-formes collaboratives intègrent généralement les éléments suivants :

- Des outils informatiques.
- Des guides ou méthodes de travail en groupe, pour améliorer la communication,
- la production, la coordination. Un service de messagerie.
- Un système de partage des ressources et des fichiers.
- Des outils de type forum, pages de discussions
- Un trombinoscope, ou annuaire des profils des utilisateurs.
- Des groupes, par projet ou par thématique.
- Un calendrier. [7]

7 Les modèles de services du Cloud

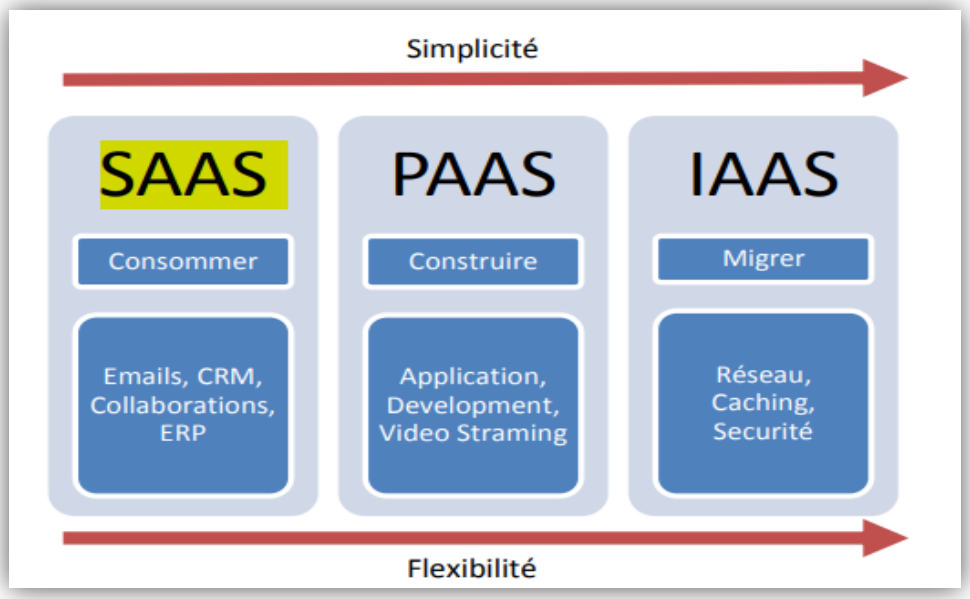


Figure I-2 Types de Services Cloud Computing [8]

Le Cloud Computing peut être décomposé en trois couches :

- Application (SaaS, Software as a Service).
- Platform (PaaS, Platform as a Service).
- Infrastructure (IaaS, Infrastructure as a Service).

La Figure ci-dessous représente les différentes couches du Cloud Computing de la couche la moins visible pour les utilisateurs finaux à la plus visible. L'infrastructure as a Service (***IaaS***) est plutôt gérée par les architectes réseaux, la couche ***PaaS*** est destinée au développeurs d'applications et finalement le logiciel comme un service (***SaaS***) est le «*Produit final* » pour les utilisateurs [13].

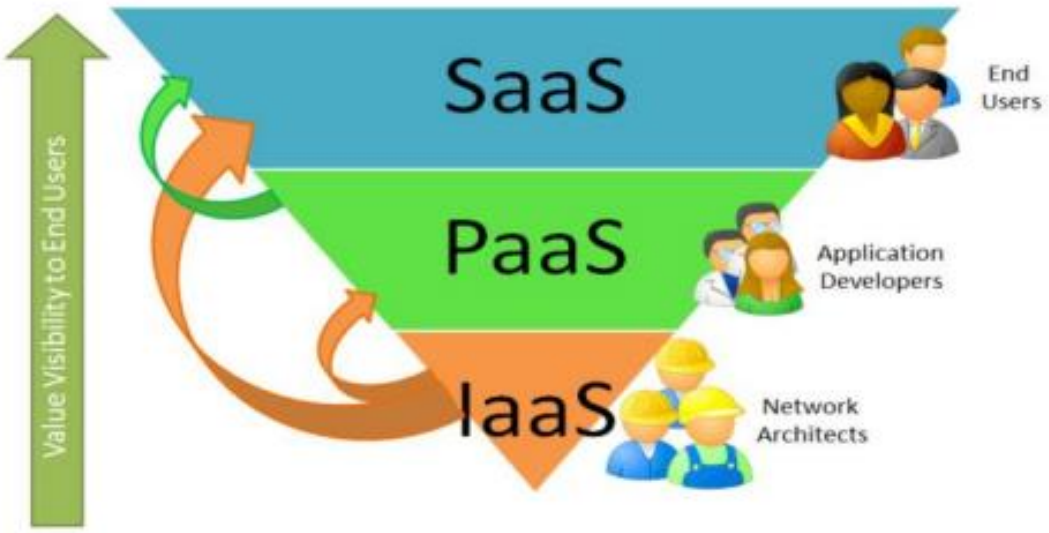


Figure I.3 Les couches du Cloud Computing [01]

7.1. frastructure as a Service (laaS)

Seul le serveur est dématérialisé. Un prestataire propose la location de composants informatiques comme des espaces de stockages, une bande passante des unités centrales et des systèmes d'exploitation, Les utilisateurs d'une ***laaS*** peuvent donc utiliser à la demande des serveurs virtuels situés dans des Datacenter sans avoir à gérer les machines physiques(coûts de gestion, remplacement de matériel, climatisation, électricité…).L'***laaS*** offre une grande flexibilité, avec une administ - ration à distance, et permet d'installer tout type de logiciel. En revanche, cette solution nécessite la présence d'un administrateur système au sein de l'entreprise. comme pour les solutions serveur classiques. Parmi les prestataires d'***laaS***, on peut citer: Amazon avec EC2 ou Orange Business Services avec Flexible Computing

7.2. Platform as a Service (PaaS)

Le matériel (*serveurs*), L'hébergement et le Framework d'application (*kit de composants logiciels structurels*) sont dématérialisés. L'utilisateur loue une plateforme sur laquelle il peut développer, tester et exécuter ses applications. Le déploiement des solutions ***PaaS*** est automatisé et évite à l'utilisateur d'avoir à acheter des logiciels ou d'avoir à réaliser des installations supplémentaires, mais ne conviennent qu'aux applications Web. Les principaux fournisseurs de ***Paas*** sont **Microsoft** avec **AZURE**, **Google** avec Google App Engine et **Orange Business Services** [11]

7.3. Software as a Service (SaaS)

Le matériel, l'hébergement, le Framework d'application et le logiciel sont dématérialisés et hébergés dans un des Datacenters du fournisseur. Les utilisateurs consomment les logiciels à la demande sans les acheter, avec une facturation à l'usage réel. Il n'est plus nécessaire pour l'utilisateur d'effectuer les installations, les mises à jour ou encore les migrations de données. Les solutions ***SaaS*** constituent la forme la plus répandue de Cloud Computing Les prestataires de solutions ***SaaS*** les plus connus sont **Microsoft-offre Office365** *(outils collaboratifs)* **Google offre Google Apps** (*messagerie et bureautique*), [12]

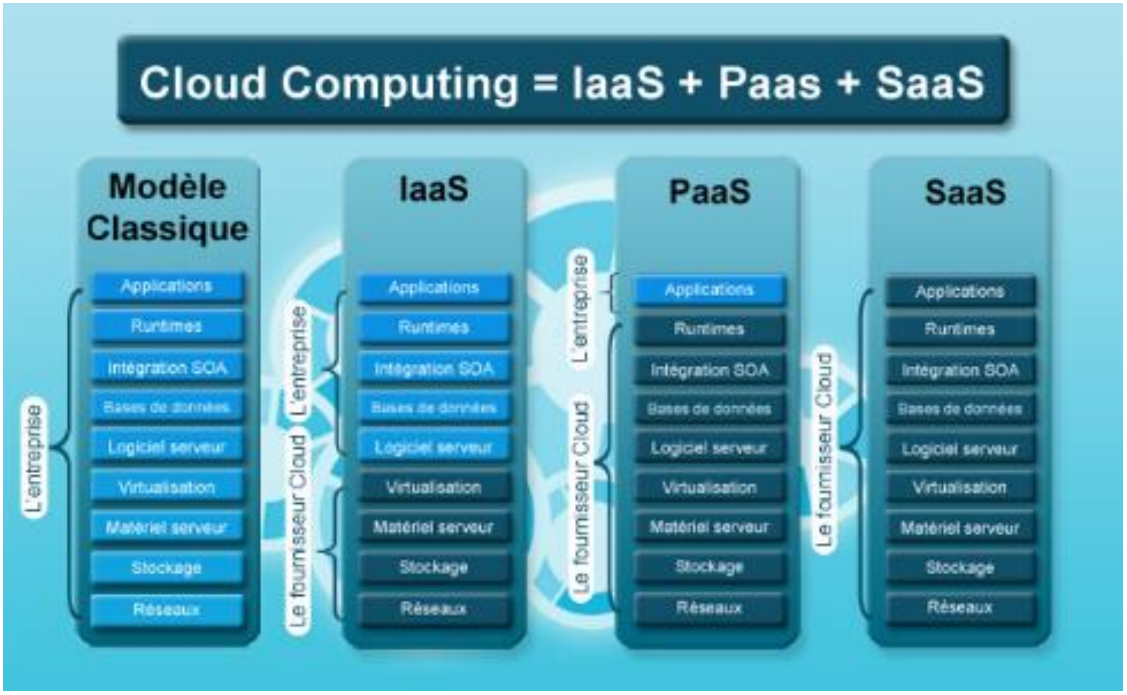


Figure I.4 *répartition des charges des principaux modèles de Cloud Computing* [10]

8 Avantages et inconvénients des services :

Les avantages et les inconvénients de ces services ce résume dans le tableau ci-dessous.

	Avantage	Inconvénient
SaaS	✓ Pas d'installation ✓ Plus de licence ✓ Migration ✓ Accessible via un abonnement	✓ Logiciel limité ✓ Sécurité ✓ Dépendance des prestataires
PaaS	✓ Pas d'infrastructure nécessaire ✓ Pas d'installation ✓ Environnement hétérogène	✓ Limitation des langages ✓ Pas de personnalisation dans la configuration des machines virtuelles
IaaS	✓ Administration ✓ Personnalisation ✓ Flexibilité d'utilisation ✓ Capacité de stockage infini	✓ Sécurité ✓ Besoin d'un administrateur système ✓ Demande pour les acteurs du Cloud des investissements très élevés

Tableau I.1 : Avantages et inconvénients des services [3] [9]

9 Les Types de Cloud:

Comme mentionné précédemment, le Cloud Computing repose sur des ressources physiques. La question qui se pose est : où se trouvent ces ressources physiques ? Serveurs, commutateurs, routeurs, solutions de stockage, etc. La réponse selon laquelle elles se trouvent "dans le nuage" n'est pas vraiment satisfaisante. En tant que consommateurs, l'abstraction est telle que nous ne pouvons pas déterminer sur quelles ressources physiques une application est hébergée. En raison de sa nature dynamique, les ressources physiques qui hébergent une application et des données dans un Cloud ne sont jamais fixes et évoluent constamment.

9.4 Le Cloud publique

Pour certaines personnes, tel que Werner Vogels *(Directeur des technologies d'Amazon),* le Cloud ne peut être que public. Un Cloud public est un service IaaS, PaaS ou SaaS proposé et hébergé par un tiers d'un ou plusieurs centres de données *Figure 2.4. Amazon*, Google et Microsoft propose un Cloud public dans lequel n'importe quel particulier ou n‟importe quelle entreprise peut y héberger ses applications, ses services ou ses données. Pour les consommateurs, il n'y a donc aucun investissement initial fixe et aucune limite de capacité. Les fournisseurs de Cloud public sont les responsables pour la gestion de la sécurité et facturent à l'utilisation et garantissent une disponibilité de services au travers des contrats SLA *(« Service Level Agreement » : document qui définit la qualité de service requise entre un prestataire et un client).*

Figure 1.5 *Un Cloud publique* ***[5]***

9.5 Le Cloud privé

Ces ressources physiques peuvent être hébergées dans une infrastructure propre à l'entreprise et étant sous son contrôle, à sa charge donc de contrôler le déploiement des applications.

La fameuse question qui se pose à ce niveau est : « ***Si un Cloud privé est réellement un Cloud ?*** ». En effet, dans le sens où, comme nous avons dit plus haut, un Cloud ne doit pas imposer de dépenses en immobilisations, l'infrastructure physique dans un Cloud privé est à la charge de l'entreprise. Le Cloud privé peut aussi désigner un Cloud déployé sur une infrastructure physique dédié

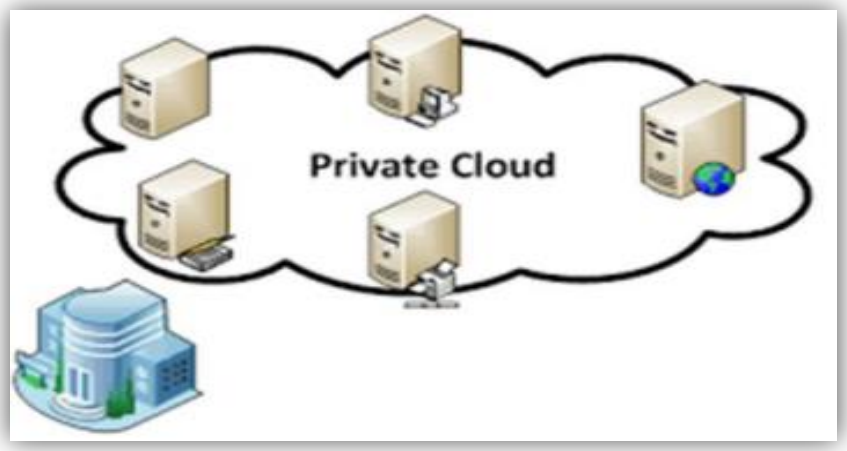


Figure 1.6 *Cloud privé* [5]

9.6 *Le Cloud hybride*

Un Cloud Hybride est l'utilisation de plusieurs Clouds, publics ou privés. Nous pouvons ainsi déporter nos applications vers un Cloud public qui consommera des données stockées et exposées dans un Cloud privé *Figure 2.6*, ou bien faire communiquer deux applications hébergées dans deux Clouds privés distincts, ou encore consommer plusieurs services hébergés dans des Cloud publiques différents

Figure 1.7** Cloud hybride **[5]

10 10. Avantages et inconvénients du Cloud Computing :

Avantages	Inconvénients
-Disponibilité et extensibilité	-Hétérogénéité
-Dynamicité	-Absence de localité
-Tolérance aux pannes	-Portage des applications
-Mutualisation des ressources	-Sécurité

Tableau I.2 : Avantages et inconvénients du Cloud Computing

11 . Les principaux acteurs du Cloud :

11.1 *Amazon*

Amazon a été la première société à proposer une plateforme du Cloud Computing avec Amazon Web Services[2] . Il s'agit d'un outil simple et facile à manipuler qui permet le développement des applications dans un environnement du Cloud Computing. Les services d'Amazon Web Services sont [14]:

- *Amazon Elastic Compute Cloud (EC2)* : il fournit un stockage basé sur les services Web.

[2] Amazon web service. Http: //aws.amazon.com/fr/.

- *Amazon Simple Storage Service (S3)* : Amazon S3 offre une simple interface de services web à utiliser pour stocker et récupérer n'importe quelle quantité de données, à tout moment, de n'importe quel endroit sur le web.
- *Amazon Cloud Front* : il permet de distribuer des objets stockés sur S3 vers un emplacement proche de l'appelant.
- *Amazon Simple DB* : il permet aux développeurs d'exécuter des requêtes sur des données structurées.
- *AWS Management Console (AWS Console)* : c'est une interface Web pour gérer et surveiller les infrastructures Amazon, incluant Amazon EC2 et Amazon S3

11.2 Cloud de Microsoft

Lors de la ***PDC*** de Novembre 2008, Microsoft annonçait l'arrivée de sa propre solution de Cloud Computing nommée Windows Azure. Cette dernière a été rendue commerciale en janvier 2010 et ne cesse de se développer au regard des annonces faites par Microsoft lors de la ***PDC*** de 28 octobre 2010.
La plate-forme Windows Azure propose depuis peu une véritable infrastructure ***laaS*** avec la notion de « *vmRole* » mais aussi une plate-forme ***PaaS*** permettant l'hébergement d'application .NET ou non, le stockage, des bases de données etc. Le Cloud de Microsoft s'est aussi des applications ***SaaS*** de la gamme live et Online service [15].

11.3 Salesforce

Salesforce.com est une société pionnière dans le domaine du ***SaaS***, elle a été créée en 1999 déjà par *Marc Benioff*. Ces dix dernières années l'entreprise n'a cessé de croître et d'enchaîner les acquisitions. En 2004 la société est introduite à la bourse de *New-York* sous le symbole boursier ***CRM***, apportant ainsi pas loin de 110 millions de dollars[3] . Basée sur la base de données *database.com* et une place de marché de logiciels, les solutions de *Salesforce.com* sont regroupées dans différentes grandes catégories : Sales Cloud, Service Cloud, *Force.com* et *Chatter Collaboration Cloud*.

- *Sales Cloud* : L'outil de ***CRM*** (*Customer Relationship Management*) par excellence disponible en plus de 25 langues et accessible depuis des appareils mobiles. Le produit fournit des outils de gestion des comptes et contacts clients, outils marketing, de ventes, plateforme de discussion.

[3] http://investor.salesforce.com/about-us/investor/investor-news/investor-news-details/2015/Salesforce-Announces-Fiscal2015-Fourth-Quarter-and-Full-Year-Results/default.aspx.

- *Service Cloud* : Un service client de nouvelle génération permettant aux entreprises d'être plus sociable et collaborative. Service Cloud propose des services rapides et réactifs intégrant tous les canaux de communication ; du centre d'appel aux réseaux sociaux.
- *Chatter Collaboration Cloud* : Est une plateforme de collaboration en temps réel reprenant un peu la forme d'un réseau social.

Salesforce.com propose également une solution **PaaS** : *Force.com* qui permet de créer des applications au moyen de Visual force *(un framework pour la création d'interfaces graphiques)* et ***Apex***, un langage de programmation propriétaire qui reprend la syntaxe de Java mais qui est plus tourné vers la gestion des bases de données.

11.4 Cloud Google

En 2008, Google a lancé son Cloud public orienté pour les services web offrant une plate-forme (***Paas***) nommée « *Google App Engine* » et permettant l'hébergement d'application Python ou Java, ainsi que des applications ***Saas*** regroupées la gamme

« Google app »

12. Conclusion

Le Cloud Computing est un concept dans laquelle l'infrastructure et les applications sont fournis à titre de services aux utilisateurs par un modèle de paiement basé sur la consommation, l'utilisateur ne paye que ce qu'il consomme grâce aux mécanismes de scalabilité et d'élasticité, tout en ayant l'impression d'avoir en permanence des ressources de calcul ou de stockage illimitées.

Le Cloud Computing est un concept jeune et en constante évolution. Une des évolutions les plus prometteuses d'après la communauté scientifique est le déploiement d'applications distribuées sur une fédération de nuage. En effet, ce mode de déploiement permet, entre autre, d'atténuer le risque de verrouillage propriétaire, de stimuler la concurrence des différents fournisseurs d'infrastructure dans le nuage, et de contrôler une partie de la confidentialité des données utilisées par les applications déployées dans le nuage.

Chapitre 2

La sécurité et la confidentialité dans le Cloud

1. Introduction :

Le Cloud Computing occupe une place de plus en plus importante dans notre vie quotidienne.

Par conséquent, la question des risques associés à cette technologie devient une préoccupation majeure et un élément essentiel à prendre en compte. En outre, le Cloud Computing permet aux utilisateurs d'accéder aux ressources et de déplacer une partie du système d'information en dehors de l'infrastructure sécurisée de l'entreprise.

Il est donc crucial de mettre en place des démarches et des mesures pour évaluer les risques et définir les objectifs de sécurité à atteindre. Dans la suite, nous allons définir les objectifs de sécurité dans le Cloud, les contraintes qui influent sur ces objectifs, ainsi que les mécanismes de sécurité utilisés dans le Cloud Computing.

2. La sécurité dans le Cloud

2.1 Sécurité physique

Le Cloud Computing, par sa nature, est lié à une forme de "*dématérialisation*" de l'hébergement (**le** *nuage*). En effet, les lieux d'hébergement du Cloud sont généralement multiples et répartis sur plusieurs centres de données. Dans le cas du Cloud public, le client n'a pas une connaissance précise des lieux d'hébergement du Cloud.

Cette particularité, qui garantit la disponibilité du Cloud, entraîne un changement significatif pour le client dans le processus de sélection de l'hébergeur. Une simple visite d'un centre de données ne suffit plus pour évaluer le niveau d'hébergement offert par un fournisseur de Cloud. Ce dernier doit être en mesure de fournir des garanties sur les conditions d'hébergement associées à son service.

Il existe plusieurs certifications et classifications reconnues et adoptées par l'ensemble des hébergeurs pour garantir la qualité de l'hébergement.

2.1.1. Accès physique

L'accès physique d'une seule personne mal intentionnée possédant une connaissance approfondie de l'infrastructure physique du Cloud Computing et de ses points sensibles peut suffire à perturber le fonctionnement du Cloud, entraînant une interruption des services et empêchant tout accès externe au Cloud. Les conséquences d'une telle intrusion peuvent être catastrophiques.

Cela peut entraîner un isolement complet ou partiel du service en cas de coupure des connexions d'accès, la perte de données en production ainsi que des données sauvegardées, sans possibilité de récupération si celles-ci sont détruites ou endommagées en dehors des données déjà sauvegardées *(externalisation et/ou réplications).*

Il existe également un risque élevé d'incendie, d'inondation, etc.

2.1.2. Contrôle et traçabilité des accès

Point critique de la sécurité physique, le contrôle des accès doit être maitrisé, que l'on soit dans un contexte de Cloud privé, public ou privé externalisé. Dans ces deux derniers cas, c'est au client final de s'assurer que les bonnes pratiques sont mises en œuvre chez son prestataire de service/opérateur de Cloud.

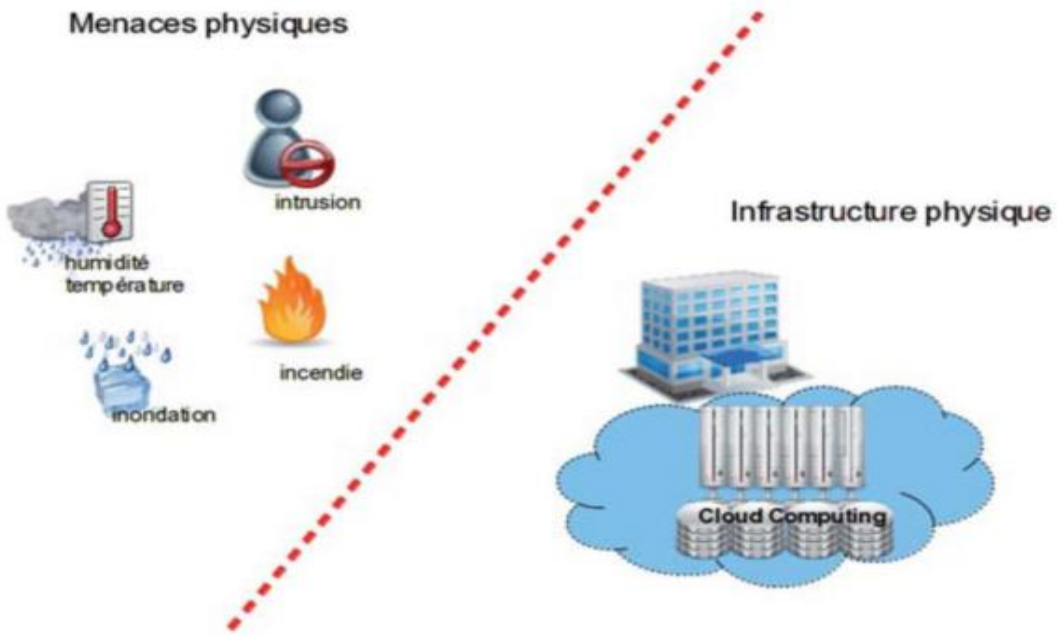
Concrètement, les va-et-vient du personnel interne et des prestataires externes *(informatique et télécoms, société de maintenance, de nettoyage, etc.)* sont nombreux dans une salle informatique ou dans les locaux techniques.

Ce flux représente une source potentielle de dysfonctionnements, volontaires ou non. Il convient de bien délimiter les zones les plus sensibles et de mettre en place des garde-fous suffisamment efficaces pour retrouver, le cas échéant, l'origine d'un incident. L'accès aux zones sensibles (*serveurs, réseau, etc.*) sera interdit et le passage dans les zones intermédiaires sera limité. Le personnel autorisé devra être informé du caractère sensible des zones dans lesquelles il est amené à intervenir. [16]

2.1.3. Redondance matérielle

L'architecture Cloud Computing doit garantir un accès au service en très haute disponibilité avec des performances optimales. La seule défaillance d'un équipement matériel peut engendrer une dégradation ou une coupure du service voire une perte de données. Pour limiter les risques d'arrêt de service liés à la défaillance d'un équipement, il est nécessaire de le redonder.

Une réplication des configurations entre les équipements peut faciliter la bonne prise en charge de la redondance et ainsi augmenter la haute disponibilité du service. La mise en œuvre d'une redondance différentielle avec une sélection d'équipements de natures différentes *(ex : différents constructeurs, composants d'origines différentes, etc.)* permet de se protéger d'un problème survenu à un équipement donné



La figure 2.1 *Sécurisation de l'environnement* [16]

De plus, la redondance des moyens de connexion, grâce à la multiplication des liaisons, des opérateurs et des chemins d'accès, améliore l'accessibilité au service en renforçant la tolérance aux panne . [16]

2.1.4. Résilience

Une catastrophe d'origine humaine ou naturelle peut avoir des conséquences majeures sur le fonctionnement du Cloud Computing, pouvant entraîner une panne totale ou partielle du service. En cas de perte totale de l'infrastructure du Cloud Computing, cela pourrait provoquer une interruption de service indéterminée et une perte de données irréversible, sans possibilité de remise en service de l'infrastructure.

Il est essentiel d'avoir une architecture de secours sur un site distant géographiquement, équipée de redondances et capable de mettre en œuvre un Plan de Continuité d'Activité (*PCA*) sans interruption du service.

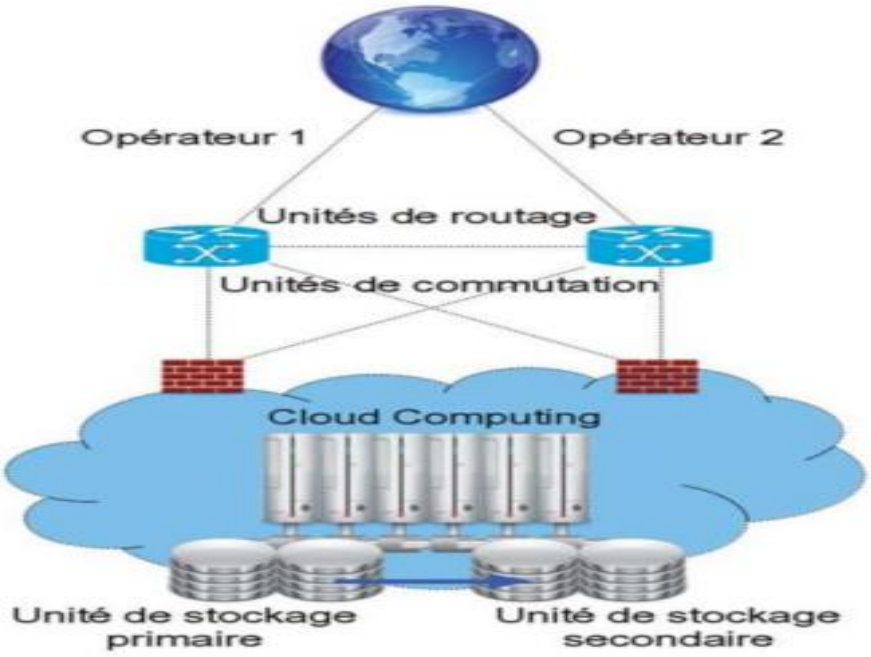


Figure 2.2 *Présentation d'une architecture mono-data center* [16]

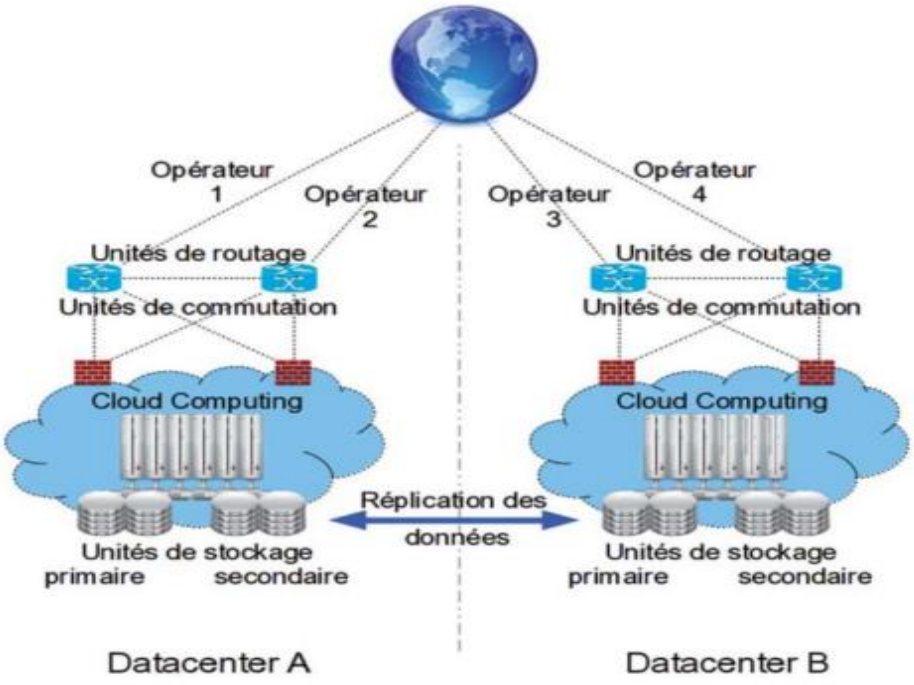


Figure 2.3 *Présentation d'une architecture multi-data center* [16]

2.2. Sécurité logique

2.2.1. La confidentialité

Consiste à s'assurer que les informations ne peuvent être divulguées à des utilisateurs non autorisés. La confidentialité des informations pourrait être assurée pendant le transfert du message, et au repos sur le serveur de stockage

2.2.2. Intégrité :

L'intégrité des données consiste à garantir qu'elles ne peuvent pas être modifiées de manière non autorisée ou non détectée. Elle est compromise lorsque des données sont modifiées de manière frauduleuse pendant leur transmission. Les systèmes de sécurité de l'information veillent généralement à préserver l'intégrité des données en plus de leur confidentialité.

2.2.3. La disponibilité

Les plateformes de Cloud Computing offrent une disponibilité robuste grâce à la redondance et à la virtualisation. ***Par exemple***, Windows Azure propose plusieurs niveaux de redondance pour assurer la disponibilité des données et des applications. Les données sont répliquées sur trois nœuds distincts pour minimiser les impacts des pannes matérielles. Les clients peuvent également profiter de l'infrastructure géographique de Windows Azure en créant un deuxième compte de stockage pour des capacités de basculement à chaud.

Les agents surveillent la santé des machines virtuelles et en cas de défaillance, le contrôleur redémarre la machine virtuelle. Les contrôleurs assurent une disponibilité continue grâce à la redondance et au basculement automatique.

2.2.4. Authentification

Incluant l'authentification de l'origine et l'authentification de l'entité. L'authentification de l'origine est un service de sécurité qui vérifie l'identité d'une entité du système qui prétend être la source originale des données reçues. L'authentification d'entité est le processus de vérification d'une revendication selon laquelle une entité système ou une ressource système a une certaine valeur d'attribut. Cet attribut est généralement l'identité de l'utilisateur.

3. La Confidentialité dans le Cloud

«Vous pouvez avoir la sécurité et ne pas avoir la confidentialité, mais vous ne pouvez jamais avoir une confidentialité sans sécurité ».

Particulièrement dans les secteurs les moins réglementés, la responsabilité et l'imputabilité de la confidentialité est souvent assigné pour IT au lieu de l'unité d'affaires qui possède les données. Donc c'est quoi la confidentialité ?

3.1 Définition

Le concept de la confidentialité varie considérablement et parfois à l'intérieur des pays, cultures, et des juridictions. Il est façonné *par les expectations du public et les interprétations du droit.* Les droits de confidentialité ou les obligations sont liées à la collecte, l'utilisation, la divulgation, le stockage et la destruction des IPIs *(Information personnelle d'un individu),* de même, il n'y a pas de consensus universel sur ce qui constitue les données personnelles, Aux fins de cette discussion, nous allons utiliser la définition adoptée par *OECD (Organization for Economic Cooperation and Development)* l'Organisation de coopération et de développement économiques : « ***Toute information relative à un individu identifié ou identifiable***». [17]

Une autre définition qui est fourni par AICPA (*the American Institute of Certified Public Accountants)* l'Institut américain des comptables publics certifiés, et par CICA *(the Canadian Institute of Chartered Accountants)* l'Institut Canadien des Comptables Agréés : « Les droits et les obligations des individus et des organisations en ce qui concerne la collecte, l'utilisation, le stockage et la divulgation des informations personnelles »

3.2 Les axes de recherches liés à la confidentialité dans le Cloud

Cette partie donne un aperçu sur les problèmes liés à la confidentialité ainsi que quelques solutions proposées dans le Cloud. Dans notre classification des problèmes, nous nous sommes basés sur les travaux de Chen et Zhao [20] et Vimercati et al [26], qui ont analysé les problèmes de la confidentialité dans le Cloud selon le cycle de vie des données.

3.2.1. Le cycle de vie des données

La gestion des informations personnelles d'un individu doit être effectuée dans le cadre des données utilisées par l'organisation, en suivant un cycle de vie des données comprenant sept phases : génération, transfert, utilisation, partage, stockage, archivage et destruction. Bien que ces phases soient présentées de manière séquentielle, les données peuvent les suivre de manière non séquentielle. Cette approche peut également être utilisée dans un système "classique" non basé sur le

Cloud, avec des mesures de sécurité différentes. La protection des informations personnelles doit prendre en compte l'impact du Cloud sur chacune de ces phases.

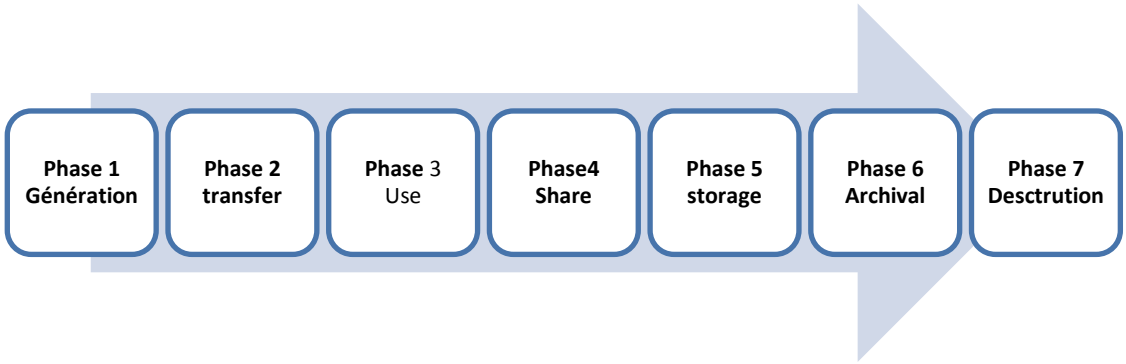


Figure 2.4 Le cycle de vie d'une donnée [16]

La définition de chacune de ces phases est comme suit :

- ***Génération de l'information :*** Ce texte traite de la production d'informations et des mesures de sécurité qui peuvent être appliquées à cette phase. Il mentionne l'utilisation d'un système de protection de la confidentialité appelé Airavat pour empêcher la fuite de données sensibles. Il fait également référence à un générateur de données pour des applications en Cloud qui est facile à étendre et à configurer grâce à son haut degré de parallélisme.

Ce dernier permet une évolutivité linéaire pour un nombre quelconque de nœuds.

- ***Le transfert*** : de données au sein d'une entreprise ne nécessite généralement pas de chiffrement, mais pour les transferts au-delà des frontières de l'entreprise, il est essentiel de garantir la confidentialité et l'intégrité des données. Ont proposé une méthode de cryptage des données dans un Cloud privé pour assurer une transmission sécurisée. soulignent les coûts élevés de la bande passante facturés par les fournisseurs de Cloud pour le chargement de données.

- ***Utilisation :*** La phase d'utilisation des données concerne les actions effectuées par un utilisateur sur les informations, où il est seul à manipuler les données. Les contrôles comprennent la gestion des droits, la surveillance des actions et les contrôles logiques au niveau des applications. Les propriétaires de données privées doivent s'assurer que l'utilisation des informations personnelles est conforme aux objectifs de collecte et vérifier si elles sont partagées avec des tiers, comme les fournisseurs de services Cloud. Des outils de gestion de la confidentialité basés sur le client ont été proposés pour permettre aux utilisateurs de contrôler le stockage et l'utilisation de leurs informations sensibles dans le Cloud. Des problèmes liés à la protection de la confidentialité, tels que

l'anonymisation des données, sont discutés lorsqu'ils sont appliqués à de grands ensembles de données.

- ***Le partage :*** Le partage des données peut rendre les autorisations de données plus complexes, car les propriétaires doivent être prudents quant à qui ils autorisent l'accès à leurs données. Il est essentiel de mettre en place des mesures de protection et des restrictions d'utilisation pour éviter tout partage non autorisé. ont proposé un cadre basé sur la responsabilité de l'information pour protéger la confidentialité des données, en identifiant les utilisateurs en fonction du type d'information utilisée et en les rendant responsables en cas d'utilisation inappropriée.

- ***Stockage*** : S'assurer que les données sont stockées, protégées et contrôlées. Plusieurs idées et suggestions sont proposées pour y parvenir, telles que l'utilisation de méthodes de contrôle d'accès, de cryptage et de surveillance des données. Une étude est mentionnée qui propose un modèle efficace et adaptable pour déboguer les erreurs de stockage des données et permettre à un utilisateur autorisé d'accéder aux données en identifiant les serveurs non conformes. Un mécanisme distribué flexible est également mentionné pour garantir l'intégrité de l'audit du stockage à l'aide d'un « code homogène et de données chiffrées par analyse distribuée ». Ce modèle permet aux utilisateurs d'auditer le stockage dans le Cloud de manière plus légère

- ***Archivage :*** La phase d'archivage consiste à transférer les données vers un support de stockage pour qu'elles y soient conservées sur une durée plus ou moins longue. Les contrôles sont principalement le chiffrement et la gestion des supports. Si les fournisseurs de services de Cloud ne fournissent pas l'archivage hors site, la disponibilité des données sera menacée. Si la durée de stockage est compatible avec les exigences d'archivage ? Sinon Cela pourrait provoquer des menaces de la disponibilité ou la confidentialité.

- ***La Destruction*** : Lorsque les données ne sont plus nécessaires, est ce qu'elles sont complètement détruite ? En raison des caractéristiques physiques du support de stockage, les données supprimées peuvent encore exister et peuvent être restaurées. Cela peut conduire à divulguer par inadvertance des informations sensibles

3.3.1 Les risques de confidentialité dans les scénarios Cloud

Les risques de confidentialité dans les scénarios Cloud sont nombreux. Parmi eux, on peut citer les

- ***Communications anonymes***

En effet, un utilisateur peut souhaiter envoyer des messages à un destinataire donné, ou espérer recevoir des réponses à ses messages. Cependant, il peut être difficile de rester anonyme dans ces situations, surtout dans le Cloud où les réponses peuvent arriver de manière différée ou presque immédiate, voire même être entrelacées avec les envois. Pour pallier à ce problème, des protocoles de communication anonymes sont proposés (Ardagna et al) , ainsi que des solutions spécifiques adaptées aux particularités des systèmes de Cloud (Jones et al) .

- ***L'exécution collaborative des requêtes***

Lors de la distribution de l'information et de la coopération dans le calcul des résultats des requêtes, il est essentiel de prendre en compte la confidentialité des données sensibles. La nécessité pour une partie de partager des informations et de coopérer avec une autre peut se présenter dans divers scénarios, allant des systèmes de bases de données distribuées classiques *(où l'administration centralisée d'une base de données est répartie)* aux systèmes de Cloud *(où les serveurs de Cloud collaborent pour intégrer leurs données et services afin de fournir aux utilisateurs des applications accessibles partout et à tout moment)*. Cette situation requiert des solutions novatrices permettant un partage sélectif des informations stockées sur plusieurs serveurs Cloud, même au-delà des frontières administratives et de l'entreprise. [22] [18]

3.3.2 Les risques de confidentialité pour l'utilisateur

Se focalisent sur la protection des identités des utilisateurs qui accèdent aux ressources dans le Cloud.

- ***Le contrôle d'accès basé sur Attribut***

Il permet aux utilisateurs d'interagir avec les serveurs de Cloud et de distribuer leurs informations sans divulguer leurs identités. Les approches traditionnelles pour l'accès aux ressources sont basées sur l'authentification des utilisateurs [24] [21] (*ces approches ne sont pas adoptées dans le Cloud où les parties qui interagissent peuvent être inconnus les uns aux autres*). Par la suite ils ont utilisé des solutions basées sur la spécification [19] [23] où les utilisateurs peuvent facilement accéder à toutes les ressources disponibles à partir (*éventuellement inconnue*) des serveurs sans avoir besoin de se souvenir de leurs mots de passe ou de gérer un compte spécifique pour chacun des serveurs sur lesquels ils accèdent.

- ***Les préférences de confidentialité de l'utilisateur***

déterminent quelles informations sont préférables à divulguer pour accéder à un service en fonction de leur sensibilité (*par exemple, préférer une carte d'identité à un passeport pour accéder à un service qui demande l'un des deux*). Ardagna et al [28] et Yao et al [29] ont abordé cet aspect. Chen et al [27] ont proposé un modèle permettant à un utilisateur d'associer un coût différent à chaque accréditation dans son portefeuille, en fonction de sa sensibilité (*c'est-à-dire que plus l'accréditation est sensible, plus le coût est élevé*), afin de minimiser le coût total d'un processus de négociation. Yao et al, [29] ont présenté un modèle de gestion de la confiance basé sur des points où un utilisateur attribue à son accréditation un score de confidentialité quantitative, et le serveur définit un crédit pour chaque accréditation que les utilisateurs peuvent posséder. Le serveur autorise un utilisateur à accéder à une ressource si la somme des crédits d'accréditations publiées atteint un seuil fixe. Enfin, Karger et al [30] ont proposé un langage basé sur la logique pour spécifier les préférences de confidentialité qui établissent un ordre partiel entre les propriétés des utilisateurs.

3.3.3 Les risques de confidentialité pour les données stockées

Depuis que la donnée est stockée dans les serveurs de Cloud (*qui ne sont pas sous le contrôle du propriétaire de donnée*) la confidentialité peut être mise en risque ! En plus, l'accès à la donnée doit être régulier (*les différentes parties peuvent avoir des privilèges d'accès différents).*

- ***La confidentialité et l'intégrité***

Dans le scénario DaaS (*Database as a Service*), le problème de la préservation de la confidentialité est abordé. Selon la solution proposée par Samarati [31], les données sont cryptées avant d'être stockées dans le serveur. Parfois, les données elles-mêmes ne sont pas sensibles, mais leur association avec d'autres informations l'est. Ainsi, le cryptage peut sembler excessif (*par exemple, la liste des noms des patients et des maladies traitées dans un hôpital peut être rendue publique, mais l'association de chaque patient à une maladie spécifique doit être protégée*).Aggarwal et al [32] et Ciriani et al [33] ont proposé des solutions pour protéger ces associations sensibles, en combinant le cryptage et la fragmentation.

- ***Accès sélectif***

Dans un scénario de Cloud, ni le propriétaire des données, ni le serveur ne peuvent appliquer la politique de contrôle d'accès. Le serveur ne peut pas directement mettre en œuvre les restrictions de contrôle d'accès car il n'a pas confiance pour les appliquer, et aussi parce que la politique régissant l'accès aux données peut dépendre du contenu des données (*qui doit rester toujours confidentiel pour le serveur).* Le propriétaire doit filtrer chaque demande d'accès, ce qui annule les avantages du Cloud. Il est donc nécessaire de concevoir un mécanisme où les données elles-mêmes imposent des restrictions sur l'ensemble des utilisateurs pouvant y accéder. Deux solutions basées sur le "***cryptage sélectif***" ont été proposées par Ciriani et al [34] et Capitani et al. [35]

Cette méthode consiste à crypter différentes parties des données en utilisant des clés différentes, et à distribuer ces clés aux utilisateurs de manière à ce que chacun puisse décrypter uniquement les éléments d'information autorisés.

4. Conclusion

Le modèle Cloud Computing offre plus de choix, de flexibilité et d'efficacité opérationnelle, permettant aux entreprises et aux individus de réaliser davantage d'économies. Pour bénéficier pleinement de ces avantages, les utilisateurs doivent avoir des garanties fiables en ce qui concerne la confidentialité et la sécurité de leurs données en ligne. Dans ce chapitre, nous avons abordé deux aspects essentiels : la sécurité physique et logique, ainsi que la confidentialité et quelques approches liées à ce domaine. Dans le prochain chapitre, nous présenterons l'aspect de l'anonymisation, qui est considéré comme l'une des techniques de confidentialité.

Chapitre 3

Les algorithmes de chiffrement a clé public et clés secrète

1. Introduction :

En général, la sécurité de l'information est un domaine très vaste qui englobe tous les aspects de la protection des données. Pour garantir sa sécurité, l'information doit être cryptée, c'est-à-dire qu'un message est écrit en utilisant des codes secrets ou des clés de cryptage. Le cryptage est principalement utilisé pour protéger les messages confidentiels et est utilisé dans de nombreux domaines tels que la défense, l'informatique et la protection de la vie privée. Il existe de nombreux algorithmes de chiffrement qui peuvent être utilisés pour crypter (et décrypter pour le destinataire) le message. Certains sont considérés comme basiques, comme le décalage des lettres de l'alphabet vers la droite ou la gauche, tandis que d'autres offrent un niveau de sécurité très élevé.

Dans ce chapitre, nous aborderons une introduction générale à la cryptographie, en commençant par un bref historique de la cryptographie avant d'examiner les grands systèmes cryptographiques modernes.

2. Définition Du Cryptage :

Le cryptage est la conversion des données d'un format lisible en un format crypté, qui ne peut être lu ou traité sans être déchiffré. C'est essentiel pour la sécurité des données et empêche qu'elles soient volées ou utilisées à des fins malveillantes. Comme le montre dans ***Figure 3.1*** [36]

Le cryptage est largement utilisé par les individus et les entreprises pour protéger les informations transmises entre le navigateur et le serveur. Un programme de cryptage, appelé aussi algorithme de cryptage, est utilisé pour créer un schéma de chiffrement inviolable par une puissance de calcul normale. [37]

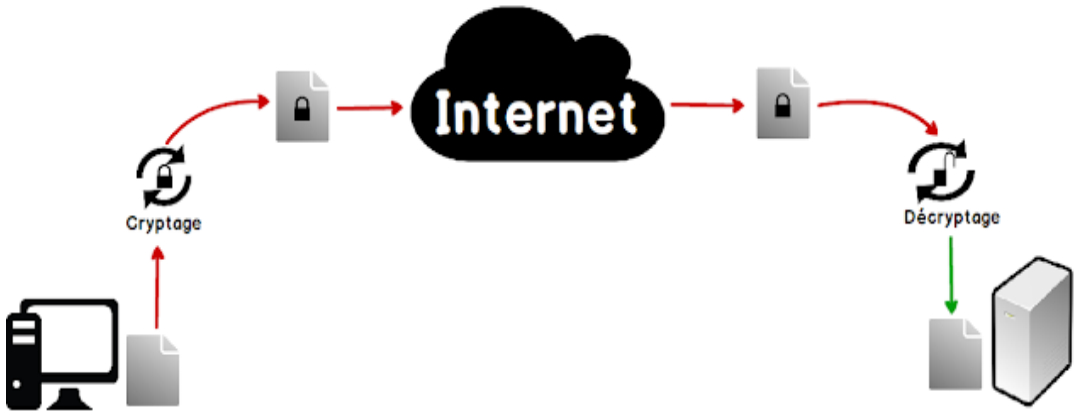


Figure 3.1 *Schéma de cryptage* [**36**]

3. Historique De Cryptage

Dans ce qui suit, nous mentionnons les étapes par lesquelles l'histoire du chiffrement est passée :

3.1.1 La Cryptographie Est Ancienne

La cryptographie est une pratique ancienne qui remonte à l'Antiquité. Elle était utilisée par de nombreuses civilisations pour remplacer les symboles dans leurs écrits. Par exemple, dans les écrits égyptiens anciens et en Mésopotamie, le cryptage était utilisé pour améliorer l'attractivité linguistique des inscriptions. Il était également utilisé pour protéger des informations sensibles, comme dans le cas d'un écrivain de cryptage en Mésopotamie qui cachait des messages dans des émaux de poterie. La cryptographie a également été utilisée pour protéger des informations militaires, comme dans le cas de l'utilisation du chiffrement à Sparte en Grèce pour empêcher les ennemis d'accéder aux messages. Un exemple célèbre de cryptage est le cryptage César utilisé par les Romains, qui consiste à déplacer les lettres de l'alphabet latin d'un certain nombre de positions pour obtenir des lettres cryptées.

3.1.2 Nouvelle Cryptographie

La cryptographie moderne a évolué avec l'avènement de l'ordinateur, devenant plus sophistiquée et avancée. Des techniques de cryptage mathématique plus puissantes ont été développées, telles que le cryptage 128 bits, pour protéger les systèmes informatiques et les appareils sensibles. Depuis les années 1990, les informaticiens ont travaillé sur des formes de cryptage modernes, comme le codage quantique, pour renforcer la sécurité des données. Les techniques de cryptage sont désormais largement utilisées dans le domaine des monnaies numériques, avec des outils tels que la fonction de hachage, les signatures numériques et la cryptographie à clé publique. L'algorithme ECDSA, qui prend en charge ***Bitcoin***, est également utilisé pour renforcer la sécurité des transactions numériques et garantir qu'elles ne sont utilisées que par les propriétaires légitimes. [38]

3.2 Terminologie De La Cryptographie

Il existe plusieurs de termes en cryptographie , notamment

- *Texte en clair* **:** c'est une information non chiffrée .
- *Texte chiffré* : appelé également cryptogramme , c'est le résultat du chiffrement du texte en clair .
- *Chiffrement* : c'est la méthode ou l'algorithme utilisé pour transformer un texte en clair en texte chiffré .

- *Déchiffrement* : c'est la méthode ou l'algorithme utilisé pour transformer un texte chiffré en texte en clair .
- *Clé* : Il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et / ou déchiffrement . Dans le cas d'un algorithme symétrique , la clef est identique 5 lors des deux opérations . Dans le cas d'algorithmes asymétriques , elle diffère pour les deux opérations
- *Cryptographie* : cette branche regroupe l'ensemble des méthodes qui permettent de chiffrer et de déchiffrer un texte en clair afin de le rendre incompréhensible pour quiconque n'est pas en possession de la clé à utiliser pour le déchiffrer
- *Cryptanalyse :* est Fart de révéler les textes en clair qui ont fait l'objet d'un chiffre ment sans connaitre la clé utilisés pour chiffrer le texte en clair .
- *Cryptologie* : il s'agit de la science qui étudie les communications secrètes . Elle est composée de deux domaines d'étude complémentaires : la cryptographie et la cryptanalyse
- *Coder , Décoder* : est une méthode ou un algorithme permettant de modifier la mise en forme d'un message sans introduire d'élément secret [39] .
- *Cryptage Homomorphe* : c'est un système de cryptage capable d'effectuer des opérations significatives sur les messages cryptés sans accéder au message d'origine [40] .

3.3 Attaques Sur Un Chiffrement

Il existe plusieurs types d'attaques sur un chiffrement. Par exemple, le système de chiffrement dépend des enregistrements transformés. Supposons que l'attaquant connaisse la forme des répétitions linéaires ainsi que la fonction du pluriel, mais pas les conditions initiales des itérations qui composent la clé du code. Il est important de distinguer entre les types d'attaques d'un adversaire et les objectifs de ces attaques. Habituellement, on distingue quatre méthodes d'attaque :

- **Une attaque par texte chiffré connu** consiste à trouver uniquement la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés.
- **Une attaque sur un clair connu** consiste à trouver une ou plusieurs clés de déchiffrement en connaissant le texte en clair correspondant.
- **L'attaque sur le texte en clair choisi** consiste à trouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, que l'attaquant a la capacité de générer à partir des textes en clair.

- **L'attaque sur le texte chiffré** choisi consiste à trouver une clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, où l'attaquant a la possibilité de générer des textes en clair. [41]

4. Types De Chiffrement

Il existe deux méthodes de cryptage, la plus courante étant le cryptage symétrique et asymétrique.. Les deux noms indiquent si la même clé est utilisée ou non pour le chiffre

4.1 Cryptographie Symétriques

Ceci est également connu sous le nom de cryptage par clé privée. La clé utilisée pour le chiffrement est la même que celle utilisée pour le déchiffrement, Comme le montre dans ***Figure 1.1*** [42] , ce qui rend cette méthode préférable aux utilisateurs individuels et aux systèmes fermés. Sinon, la clé doit être envoyée au destinataire. Cependant, cela augmente le risque d'être piraté s'il est intercepté par un tiers, tel que des pirates. Mais cette méthode est plus rapide que la méthode asymétrique [43]

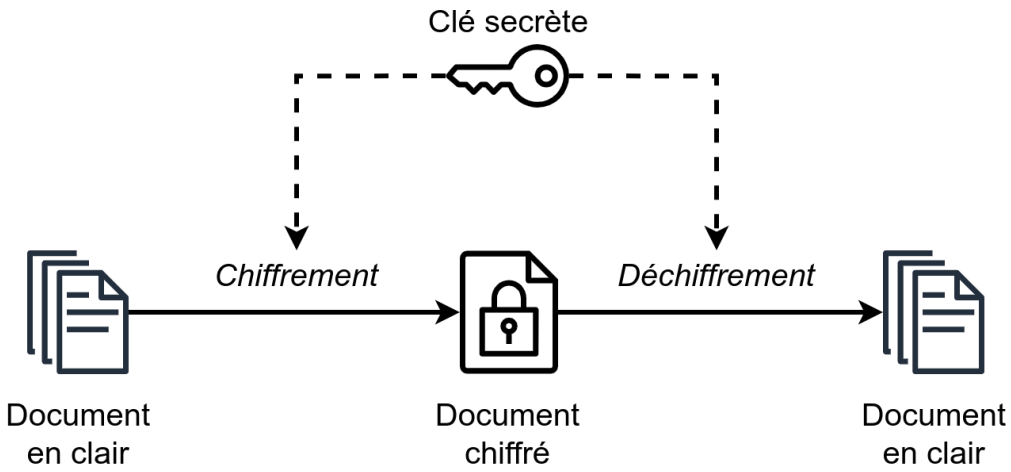


Figure 3.2: *Cryptographie Symétriques* [42]

4.2 Cryptographie Asymétrique

Cette méthode utilise deux clés différentes, une publique et une privée, qui sont liées arithmétiquement. Les deux clés sont fondamentalement de grands nombres, et elles sont liées mais ne sont pas identiques, d'où le nom "asymétrique". La clé privée est gardée secrète par le propriétaire et la clé publique est soit partagée entre les destinataires autorisés, soit rendue publique. Les données chiffrées avec la clé publique du destinataire ne peuvent être déchiffrées qu'avec la clé privée correspondante.[43] Comme le montre dans ***Figure 3.3*** [42]

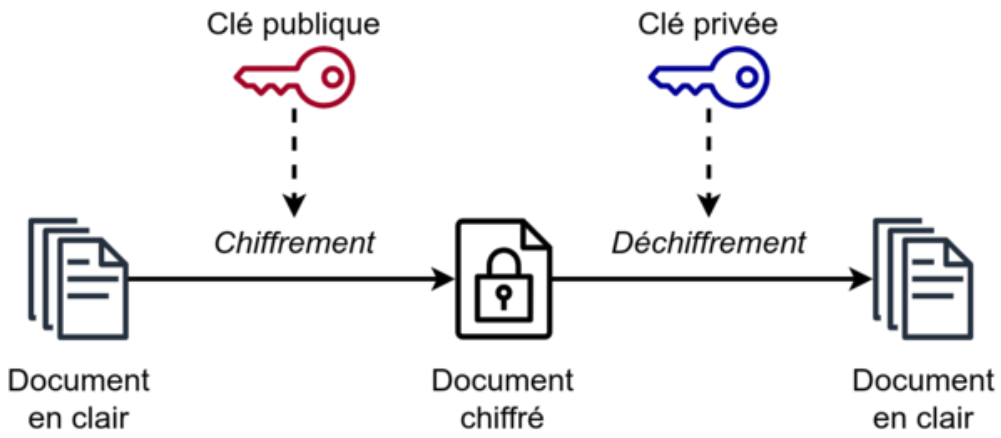


Figure 3.3: *Cryptographie Asymétrique* [42]

5. La Cryptographie Homomorphe

Dans ce qui suit, nous donnerons la définition cryptographie d'homomorphe, son histoire, et ses types :

5.1 Définition De La Cryptographie Homomorphe

C'est une forme de cryptage avec certaines propriétés algébriques qui permet la création d'opérations arithmétiques spéciales sur le texte chiffré pour obtenir une correspondance cryptée lors du décryptage à la suite d'opérations effectuées sur le texte d'origine, ce qui signifie que le décodage du résultat de ce processus est sur données non cryptées. Cette fonctionnalité permet d'attribuer des opérations arithmétiques à un agent externe, sans accéder à ses données ou résultats Le service de Cloud Computing peut être appelé pour effectuer des calculs en garantissant la confidentialité des données traitées [44].

5.2 L'histoire de la Cryptographie Homomorphe

Les schémas de chiffrement homomorphes ont été développés en utilisant diverses approches. En particulier, les schémas de chiffrement entièrement homomorphes sont souvent classés en générations correspondant à l'approche sous-jacente.

Avant le FHE, le problème de la construction d'un schéma de chiffrement entièrement homomorphe a été proposé pour la première fois en 1978, moins d'un an après la publication du schéma RSA. Pendant plus de 30 ans, on ne savait pas s'il existait une solution.

- **FHE De Première Génération**

Craig Gentry, utilisant la cryptographie basée sur un réseau, a décrit la première construction plausible d'un schéma de chiffrement entièrement homomorphe.
Le schéma de Gentry prend en charge à la fois les opérations d'addition et de Multiplication sur les textes chiffrés, à partir desquels il est possible de construire des circuits pour effectuer des calculs arbitraires. La construction commence à partir d'un peu schéma de cryptage homomorphe, qui se limite à évaluer des polynômes de faible degré sur des données cryptées ; il est limité car chaque texte chiffré est bruyant dans un certain sens, et ce bruit augmente à mesure que l'on ajoute et multiplie les textes chiffrés, jusqu'à ce que finalement le bruit rende le texte chiffré résultant indéchiffrable. Gentry montre ensuite comment modifier légèrement ce schéma pour le rendre amorçable, c'est-à-dire capable d'évaluer son propre circuit de déchiffrement puis au moins une opération de plus. Enfin, il montre que tout schéma de chiffrement amorçable quelque peu homomorphe peut être converti en un chiffrement entièrement homomorphe grâce à une auto-incorporation récursive. Pour le schéma "bruyant" de Gentry, la procédure d'amorçage "rafraîchit" efficacement le texte chiffré en lui appliquant la procédure de déchiffrement de manière homomorphe, obtenant ainsi un nouveau texte chiffré qui chiffre la même valeur qu'auparavant mais a moins de bruit. En "*rafraîchissant*" périodiquement le texte chiffré chaque fois que le bruit devient trop important, il est possible de calculer un nombre arbitraire d'additions et de multiplications sans trop augmenter le bruit. Gentry a basé la sécurité de son schéma sur la dureté supposée de deux problèmes: certains problèmes du pire des cas sur des treillis idéaux et le problème de la somme des sous-ensembles clairsemés *(ou defaible poids)*. Doctorat de Gentry. la thèse fournit des détails supplémentaires. L'implémentation Gentry-Halevi du cryptosystème original de Gentry a signalé un temps d'environ 30 minutes par opération de bit de base. Des travaux de conception et de mise en œuvre approfondis au cours des années suivantes ont amélioré ces premières implémentations de plusieurs ordres de grandeur en termes de performances d'exécution.
En 2010, Marten van Dijk, Craig Gentry, Shai Halevi et Vinod Vaikuntanathan ont présenté un deuxième schéma de chiffrement entièrement homomorphe, qui utilise de nombreux outils de construction de Gentry, mais qui ne nécessite pas de réseaux idéaux. Au lieu de cela, ils montrent que la composante quelque peu homomorphe du schéma idéal basé sur un réseau de Gentry peut être remplacée par un schéma quelque peu homomorphe très simple qui utilise des nombres entiers. Le schéma est donc conceptuellement plus simple que le schéma de réseau idéal de Gentry, mais a des propriétés similaires en ce qui concerne les opérations homomorphes et l'efficacité. La composante quelque peu homomorphe des travaux de Van Dijk et al.

est similaire à un schéma de cryptage proposé par Levieil et Naccache en 2008,et également à celui qui a été proposé par Bram Cohen en 1998.
Cependant, la méthode de Cohen n'est même pas additivement homomorphe. Le schéma Levieil – Naccache ne prend en charge que les additions, mais il peut être modifié pour prendre également en charge un petit nombre de multiplications. De nombreux raffinements et optimisations du schéma de Van Dijk et al. ont été proposés dans une séquence d'œuvres de Jean-Sébastien Coron, TancrèdeLepoint, Avradip Mandal, David Naccache et Mehdi Tibouchi. Certains de ces travaux comprenaient également des implémentations des schémas résultants

- **FHE De Deuxième Génération**

Les cryptosystèmes homomorphes de cette génération sont dérivés de techniques développées à partir de 2011-2012 par ZvikaBrakerski, Craig Gentry, Vinod Vaikuntanathan et d'autres. Ces innovations ont conduit au développement de cryptosystèmes quelque peu et totalement homomorphes beaucoup plus efficaces. Ceux-ci inclus:

- Le schéma Brakerski-Gentry-Vaikuntanathan *(BGV, 2011*), s'appuyant sur les techniques de Brakerski-Vaikuntanathan
- Le schéma basé sur NTR0U de Lopez-Alt, Tromer et Vaikuntanathan *(LTV, 2012*)
- Le schéma Brakerski/Fan-Vercauteren *(BFV, 2012*), s'appuyant sur le cryptosystème invariant à l'échelle de Brakerski ; Le schéma basé sur NTRU de Bos, Lauter, Loftus et Naehrig (*BLLN, 2013*), s'appuyant sur le cryptosystème invariant d'échelle de LTV et de Brakerski ; La sécurité de la plupart de ces schémas est basée sur la dureté du problème d'apprentissage (en anneau) avec erreurs (RLWE), à l'exception des schémas LTV et BLLN qui reposent sur une variante trop étendue du problème de calcul NTRU. Cette variante NTRU s'est ensuite révélée vulnérable aux attaques de réseau de sous-champs, c'est pourquoi ces deux schémas ne sont plus utilisés dans la pratique. Tous les cryptosystèmes de deuxième génération suivent toujours le plan de base de la construction originale de Gentry, à savoir qu'ils construisent d'abord un cryptosystème quelque peu homomorphe, puis le convertissent en un cryptosystème entièrement homomorphe en utilisant le bootstrap. Une caractéristique distinctive des cryptosystèmes de deuxième génération est qu'ils présentent tous une croissance beaucoup plus lente du bruit lors des calculs homomorphes. Des optimisations supplémentaires par Craig Gentry, Shai Halevi et Nigel Smart ont abouti à des cryptosystèmes avec une complexité asymptotique presque optimale : effectuer des opérations sur des données chiffrées avec un paramètre de sécurité a une complexité de seulement . Ces optimisations s'appuient sur les techniques Smart-

Vercauteren qui permettent de regrouper de nombreuses valeurs de texte en clair dans un seul texte chiffré et de fonctionner sur toutes ces valeurs de texte en clair de manière SIMD. Bon nombre des avancées de ces cryptosystèmes de deuxième génération ont également été portées sur le cryptosystème sur les nombres entiers. Une autre caractéristique distinctive des schémas de deuxième génération est qu'ils sont suffisamment efficaces pour de nombreuses applications, même sans invoquer l'amorçage, fonctionnant plutôt en mode FHE nivelé.

- **FHE De Troisième Génération**

En 2013, Craig Gentry, Amit Sahai et Brent Waters (GSW) ont proposé une nouvelle technique pour construire des schémas FHE qui évite une étape coûteuse de "relinéarisation" dans la multiplication homomorphe. ZvikaBrakerski et Vinod Vaikuntanathan ont observé que pour certains types de circuits, le cryptosystème GSW présente un taux de croissance du bruit encore plus lent, et donc une meilleure efficacité et une sécurité renforcée. Jacob Alperin-Sheriff et Chris Peikert ont ensuite décrit une technique de bootstrap très efficace basée sur cette observation. Ces techniques ont été encore améliorées pour développer des variantes en anneau efficaces du cryptosystème GSW : FHEW (2014) et TFHE (2016). Le schéma FHEW a été le premier à montrer qu'en actualisant les textes chiffrés après chaque opération,il est possible de réduire le bootstrapping. temps à une fraction de seconde. FHEW a introduit une nouvelle méthode pour calculer les portes booléennes sur des données cryptées qui simplifie grandement le bootstrap et a mis en œuvre une variante de la procédure de bootstrap. L'efficacité de FHEW a été encore améliorée par le schéma TFHE, qui implémente une variante en anneau de la procédure d'amorçage en utilisant une méthode similaire à celle de FHEW

- **FHE de quatrième génération**

En 2016, Cheon, Kim, Kim et Song (CKKS) ont proposé un schéma de chiffrement homomorphe approximatif qui prend en charge un type spécial d'arithmétique à virgule fixe communément appelée arithmétique à virgule flottante par blocs. Le schéma CKKS comprend une opération de remise à l'échelle efficace qui réduit un message chiffré après une multiplication. À titre de comparaison, un tel redimensionnement nécessite un amorçage dans les schémas BGV et BFV. L'opération de remise à l'échelle fait du schéma CKKS la méthode la plus efficace pour évaluer les approximations polynomiales et constitue l'approche préférée pour la mise en œuvre d'applications d'apprentissage automatique préservant la confidentialité. Le schéma introduit plusieurs erreurs d'approximation, à la fois non déterministes et déterministes, qui nécessitent un traitement spécial dans la pratique. Un article de 2020 de Baiyu Li et Daniele Micciancio traite des attaques passives

contre CKKS, suggérant que la définition standard IND-CPA peut ne pas être suffisante dans les scénarios où les résultats de décryptage sont partagés. Les auteurs appliquent l'attaque à quatre bibliothèques de chiffrement homomorphes modernes (HEAAN, SEAL, HElib et PALISADE) et rapportent qu'il est possible de récupérer la clé secrète à partir des résultats de déchiffrement dans plusieurs configurations de paramètres. Les auteurs proposent également des stratégies d'atténuation pour ces attaques et incluent une divulgation responsable dans l'article suggérant que les bibliothèques de chiffrement homomorphe ont déjà mis en œuvre des atténuations pour les attaques avant que l'article ne soit rendu public. De plus amples informations sur les stratégies d'atténuation mises en œuvre dans les bibliothèques de chiffrement homomorphe ont également été publiées [45].

5.3 Les Types De Chiffrement Homomorphe

Voici les types de cryptage [46] :

5.3.1 Chiffrement Homomorphe Additif

Dans un contexte de chiffrement additif, un serveur distant pourra retourner le résultat d'une opération d'addition sur les messages en clair en faisant le calcul sur des messages chiffrés, sans disposer de la clé secrète.

$$\mathbf{ENC}(\mathbf{X}+\mathbf{Y}) = \mathrm{Enc}(\mathbf{X}) + \mathrm{Enc}(\mathbf{Y})$$
$$\prod_{i=1}^{n} \mathrm{ENC}\,(\mathbf{m_i}) = \mathrm{ENC}\left(\sum_{i=1}^{n} m_i\right) \qquad (1.1)$$

En d'autres termes, soient: Encp une fonction de chiffrement à clé publique p. Decs une fonction de déchiffrement à clé secrète s. Alors :

$$\mathrm{Dec}_s\left(\mathrm{Enc}_\mathrm{p}(\mathbf{m}) \times \mathrm{Enc}_\mathrm{p}(\mathbf{n})\right) = \mathbf{m} + \mathbf{n} \qquad (1.2)$$

Les algorithmes de chiffrement qui possèdent cette caractéristique de chiffrement homomorphe additif sont : Paillier et Goldwasser-Micali.

- ***Chiffrement Homomorphe De Paillier***

Le système de cryptage de Paillier est un système de cryptage asymétrique développé par Pascal Paillier en 1999. Ce système de cryptage possède la plus grande bande passante, également appelée taux d'expansion, qui est le rapport entre la longueur du texte en clair et la longueur du texte chiffré.

- ***Le chiffrement homomorphe de GM***

Est un algorithme asymétrique de cryptographie à clé publique développé par Shafi Goldwasser et Silvio Micali en 1982. Ce système de chiffrement repose sur la notion de chiffrement probabiliste, ce qui signifie qu'il intègre de l'aléa dans le processus de chiffrement pour garantir sa sécurité. Cependant, le schéma de GM basé sur la difficulté du problème de la résiduosité quadratique n'est pas très efficace, car les textes chiffrés peuvent être beaucoup plus longs que les textes d'origine, parfois des centaines de fois plus longs.

5.3.2 chiffrement homomorphe multiplicatif

Par analogie avec ce qui précède, un système basé sur le chiffrement homomorphe multiplicatif permet de n'effectuer que des produits sur les clairs, sans disposer de la clé secrète. Définition : Un chiffrement homomorphe est multiplicatif si :

$$\begin{aligned} \mathbf{ENC}(\mathbf{X}+\mathbf{Y}) &= \mathrm{Enc}(\mathbf{X}) + \mathrm{Enc}(\mathbf{Y}) \\ \prod_{i=1}^{n} \mathrm{ENC}\left(\mathbf{m_i}\right) &= \mathrm{ENC}\left(\prod_{i=1}^{n} m_i\right) \end{aligned} \qquad (1.3)$$

En d'autres termes, soient : EncP une fonction de chiffrement à clé publique p. DecS une fonction de déchiffrement à clé secrète s.Alors:

$$\mathrm{Dec}_s\left(\mathrm{Enc}_\mathrm{p}(\mathbf{m}) \times \mathrm{Enc}_\mathrm{p}(\mathbf{n})\right) = \mathbf{m} \times \mathbf{n} \qquad (1.4)$$

Parmi les algorithmes Homomorphes permettant ce type d'opération, nous citons RSA et El Gamal.

- ***Chiffrement Homomorphe De RSA***

Le premier système à clé publique à être proposé fut celui de Ronald Rivest, Adi Shamir et Leonard Adleman, connu sous le nom de RSA. Cet algorithme a été décrit en 1978. Parmi tous les systèmes cryptographiques asymétriques actuels, RSA est considéré comme l'un des plus solides, voire le plus solide. Il a résisté à des années de cryptanalyse intensive et est encore considéré comme suffisamment robuste pour protéger les échanges bancaires et d'autres données critiques. Ce niveau de sécurité repose sur la difficulté de factoriser de grands nombres. Retrouver le texte en clair à partir d'une clé et du texte chiffré est supposé être équivalent à la factorisation du produit des deux nombres premiers.

- ***Chiffrement Homomorphe D'el GamalLe***

crypto système d'El Gamal est une méthode de cryptographie à clé publique inventée par Taher ElGamal en 1985. Sa sécurité repose sur la difficulté de calculer le logarithme discret.

5.3.3 Chiffrement Complètement Homomorphe

Avec le chiffrement partiellement homomorphe, nous pouvons effectuer tous les types de calculs sur les données chiffrées stockées dans le Cloud sans les déchiffrer. L'utilisation de ce chiffrement complètement homomorphe est un élément clé de la sécurité du Cloud. Contrairement à ce qui se fait généralement, il serait possible de sous-traiter des calculs sur des données confidentielles à des serveurs situés dans le Cloud tout en conservant la clé secrète permettant de déchiffrer le résultat du calcul.

En 2014, le chiffrement homomorphe devient très prometteur : la Commission européenne encourage dans son dernier appel à projets ICT l'utilisation du chiffrement homomorphe dans des applications d'ici 2020. Le projet HEAT a réuni avec succès des chercheurs de premier plan sur ce sujet en Europe (des universités de Leuven, Bristol et du Luxembourg) ainsi que des partenaires industriels spécialisés en cryptographie avancée (Crypto Experts, NXP et Thales) intéressés par le chiffrement homomorphe.

Définition :
Un système de chiffrement complètement homomorphe est un système cryptographique qui permet d'effectuer des calculs sur les données chiffrées sans les déchiffrer. Formellement, si ci *(respectivement c2)* est le chiffrement de m1 *(respectivement m2),* il existe deux opérations et

$$\mathrm{Dec}\,(c_1 \square c_2) = \mathrm{Dec}\,(c_1) \odot \mathrm{Dec}\,(c_2) = m_1 \odot m_2$$

---

a été initié par Craig Gentry, ensuite DGHV une nouvelle version de son algorithme appliquée sur les entiers a vu le jour en 2010.

- ***Chiffrement De Graig Gentry***

Le premier système de cryptage entièrement intégré a été décrit par Gentry en 2009, utilisant des réseaux de triangles de polynômes. La sécurité de ce système repose sur les réseaux de triangles. Pour crypter un message, l'idée est d'ajouter du bruit ou de petites erreurs. La clé secrète permet de supprimer ce bruit, à condition qu'il ne soit pas trop important. Les opérations homomorphes effectuées affectent également ce bruit, le faisant augmenter. Le déchiffrement du message ne peut être effectué que si le bruit initial est choisi très petit. Pour contourner cette contrainte sur le nombre d'opérations,et lorsque le bruit devient trop important, Gentry applique une méthode de "*bootstraping*" ou de démarrage. Si le déchiffrement est suffisamment efficace, nous pouvons changer la clé publique pour réduire le bruit. Nous commençons par utiliser

une clé initiale, puis lorsque le bruit devient trop important, nous utilisons une deuxième clé pour rechiffrer le même message. Le bruit ou l'erreur e dans un triangle de boucle R est défini par : e = k|£|cR. Le message est crypté en ajoutant ce bruit au message.

5.3.4 Chiffrement Partiellement Homomorphe

On dira d'un crypto système qu'il est partiellement homomorphe lorsque son espace fonctions évaluables est une restriction de l'espace des fonctions calculables. Par exemple, le crypto système à clé publique RSA est partiellement homomorphe vis-à-vis de la multiplication. En effet, si (x,y) est un couple d'entiers, on a alors l'égalité suivante

$$\begin{aligned} \varepsilon(\mathbf{x}) \cdot \varepsilon(\mathbf{y}) &= \mathbf{x}^{\mathrm{e}} \cdot \mathbf{y}^{\mathrm{e}} \bmod \mathbf{N} \\ &= (\mathbf{x} \cdot \mathbf{y})^{\mathrm{e}} \bmod \mathbf{n} \\ &= \varepsilon(\mathbf{x}, \mathbf{y}) \end{aligned} \tag{1.5}$$

- **Chiffrement De Benaloh** L'algorithme de Benaloh est détaillé ci - dessous :
  Algorithme Benaloh : Génération des clés : r la taille des blocs , p et q deux grand nombres premiers tel que :
  $\mathrm{r} \mid \mathrm{p}-1, \mathrm{pgcd}\left(\mathrm{r}, \frac{p-1}{r}\right) = 1$ et $\mathrm{pgcd}(\mathrm{r}, \mathrm{q}-1) = 1$
  Calculer : $\mathrm{N} = \mathrm{pq}$
  Choisir : $\mathrm{y} \in Z_N^*$ au hasard tel que : $\frac{(p-1)(q-1)}{r} \bmod N \neq 1$
  La clé publique : $(\mathrm{y}, \mathrm{r}, \mathrm{N})$
  La clé privée : (p, q)
  Chiffrement : pour $m£Z_{\mathrm{r}}$ Choisir $u \in Z_N^*$ au hasard Calculer $\mathrm{c} = \mathrm{X}_{\mathrm{m}}^{\mathrm{m}}\mathrm{u}^{\mathrm{r}} \bmod \mathrm{N}$
  Déchiffrement: doit faire une recherche exhaustive pour trouver quel if $\{0, \ldots, \mathrm{p}-1\}$
  Vérifie : $\left(\mathrm{y}^{-1}\mathrm{c} \bmod \mathrm{N}\right)^{\frac{(p-1)(q-1)}{r}} \bmod \mathrm{N} = 1$
  le chiffrement de Benaloh est homomorphe pour l'addition :
  $\mathrm{Dec}_{\mathrm{sk}}\ ((C_1 * C_2) \bmod N) = (m_1 + m_2) \bmod r$

6. L'utilisation du chiffrement homomorphe

Le chiffrement homomorphe est essentiel car il permet d'effectuer des opérations sur des données chiffrées. Cela signifie que le traitement des données peut être délégué sans avoir à faire confiance à un tiers pour sécuriser les données de manière adéquate. Sans la clé de déchiffrement appropriée, les données d'origine restent inaccessibles. Les applications potentielles de la cryptographie homomorphe sont

nombreuses dans la vie réelle. Sa pertinence se manifeste dans de nombreux défis commerciaux majeurs auxquels font face les entreprises de tous les secteurs.

6.1 La vie privée des consommateurs dans la publicité

Bien que la publicité soit souvent indésirable, elle peut être utile lorsqu'elle est adaptée aux besoins des utilisateurs, par exemple à travers des systèmes de recommandation ou des publicités locales. Cependant, de nombreux utilisateurs sont préoccupés par la confidentialité de leurs données, telles que leurs préférences ou leur emplacement géographique. Plusieurs approches ont été proposées pour résoudre ce problème. Jeckmans et d'autres ont proposé un scénario dans lequel l'utilisateur souhaite recevoir des recommandations de produits. Le scénario est conçu autour d'un réseau social où les recommandations sont basées sur les goûts des amis de l'utilisateur tout en garantissant la confidentialité. Le système proposé met en œuvre un processus de clonage homomorphe pour permettre à l'utilisateur de recevoir des recommandations d'amis sans révéler l'identité du recommandateur. Armknecht et Strufe ont proposé un système de recommandation où l'utilisateur reçoit des recommandations chiffrées sans que le système ne connaisse le contenu. Ce système repose sur un modèle de clonage hautement symétrique mais très efficace développé à cet effet. Cela permet de calculer une fonction de ciblage publicitaire pour chaque utilisateur tout en gardant la publicité chiffrée. Dans une autre méthode de publicité ciblée, le téléphone portable envoie la localisation de l'utilisateur au fournisseur de services, qui envoie à son tour des publicités personnalisées à l'utilisateur, telles que des coupons de réduction pour les magasins à proximité. Bien sûr, cela permet au fournisseur de services de surveiller tous les aspects des habitudes et préférences de l'utilisateur. Cependant, ce problème peut être résolu en utilisant un chiffrement homomorphe à condition que les publicités proviennent d'un tiers (ou de plusieurs parties) et qu'il n'y ait pas de collusion avec le fournisseur de services.

6.2 Système Médical

Afin d’améliorer la qualité des soins de santé, le dossier de santé électronique est la meilleure approche pour maintenir les dossiers des patients. Un dossier médical électronique (DSE) est un enregistrement des détails médicaux d’un patient au format numérique. Cette méthode présente plusieurs avantages par rapport aux documents 16 papier. Le dossier de santé électronique nous aide à conserver un grand nombre de dossiers, y compris l’efficacité et la mise à jour rapide des dossiers avec précision . Ce système de dossiers médicaux comprend des résultats de laboratoire, des listes de médicaments, des tests de diagnostic, des évaluations physiques, des notes sur les antécédents et des outils de gestion de la santé des patients. Un même dossier patient

peut être consulté par plusieurs utilisateurs. Ces données sont stockées dans le cloud public. Il existe de nombreux problèmes de sécurité dans le cloud computing et c'est pourquoi la confidentialité doit être prise en considération. En raison du nom, de l'adresse et d'autres dossiers médicaux du patient, il peut y avoir un risque de vol, d'accès non autorisé et de violation de données. Ces registres doivent être tenus efficacement. L'accès de tiers à ces enregistrements peut être autorisé en cryptant le message d'origine à l'aide d'algorithmes de cryptage. Les unités de santé peuvent nécessiter que certaines opérations soient effectuées sur des données stockées dans le cloud et se rapprochent des fournisseurs de services cloud. Dans ce cas, les propriétaires doivent divulguer les données originales. Il doit donc y avoir une approche pour effectuer des opérations sans exposer le message d'origine. Le chiffrement symétrique permet d'effectuer le calcul sur un texte chiffré, puis les résultats des opérations effectuées par le propriétaire peuvent être déchiffrés. Désormais, le propriétaire des données peut recevoir le même message que s'il était exécuté sur deux textes en clair. La propriété homomorphe fournit un moyen de préserver les données d'origine en permettant à un tiers d'effectuer des opérations sur les données chiffrées [46] .

6.3 Secteur Financier

Dans le domaine financier, la confidentialité des informations est essentielle pour les clients et les entreprises. Les données et les calculs effectués sur ces données doivent donc rester privés. Par exemple, les données sur les sociétés, les cours des actions, la performance et l'inventaire sont cruciales pour les décisions d'investissement. Ces informations peuvent être diffusées en continu pour aider à prendre des décisions commerciales basées sur les données les plus récentes.

Les entreprises cherchent à protéger leurs modèles prédictifs de performance des actions, développés à partir de recherches coûteuses menées par des analystes financiers. Pour rester compétitives et protéger leurs investissements, elles préfèrent garder ces modèles confidentiels. L'utilisation d'un cryptage FHE permet de traiter certaines fonctions de manière privée. Par exemple, un client peut télécharger une version cryptée d'une fonction dans le cloud, où les évaluations sont effectuées sur des données chiffrées. Les données de streaming peuvent être cryptées avec la clé publique du client, puis traitées dans le cloud. Une fois le traitement effectué, le cloud renvoie la sortie chiffrée au client [47] .

7. Conclusion

Au cours de ce chapitre, nous avons abordé de manière générale quelques concepts fondamentaux et l'évolution de la cryptographie, les divers types de chiffrement, les attaques possibles ainsi que le chiffrement homomorphe et ses applications et son évolution.

Chapitre 4

Cryptage des données a base de l'acide désoxyribonucléique (ADN).

1. Introduction :

La protection des informations est l'une des questions les plus critiques dans le système de communication en raison de la croissance importante des applications de transport. Le cryptage est mis en avant comme une technique importante pour garantir la confidentialité et l'intégrité des données, ainsi que l'authentification dans la livraison des informations. Il existe deux catégories de systèmes de cryptage : le cryptage symétrique et le cryptage asymétrique, en fonction de la clé utilisée. Le cryptage symétrique implique une clé partagée entre l'expéditeur et le destinataire, tandis que le cryptage asymétrique utilise des clés publiques et privées pour le cryptage et le décryptage. Le masquage des données implique une série de procédures pour incorporer les informations dans différents modèles de supports tels que l'image, le texte, l'image et la vidéo en apportant des modifications subtiles au support d'information. L'utilisation de chaînes d'ADN comme support d'information pour masquer les données est suggérée en raison de ses caractéristiques uniques qui rendent difficile la distinction entre les chaînes originales et les fausses. [51]

La cryptographie de l'ADN introduit une technique améliorée pour stocker une grande quantité de données dans le petit fragment d'ADN et assure la sécurité et l'intégrité des données. Une efficacité élevée et un stockage massif de données peuvent être observés dans la structure de la molécule d'ADN. La cryptographie de l'ADN est généralement définie comme le fait de cacher des données dans la séquence d'ADN. Les avantages de cette cryptographie sont une vitesse de calcul améliorée, agit comme un support avec une grande capacité de transmission de données et une consommation d'énergie moindre. Les quatre types de bases azotées de l'ADN sont l'adénine (A) et la thymine (T) ou la cytosine (C) et la guanine (G) dans la séquence d'ADN. Les motifs de codage d'ADN les plus simples pour encoder les 4 bases nucléotidiques (A, T, C, G) sont au moyen de 4 chiffres : 0(00), 1(01), 2(10), 3(11) qui sont représentés dans le tableau ci-dessous. [48][53]

Base d'ADN	Code
A	00
C	01
G	10
T	11

Tableau 4.1 : Codage basé sur l'ADN

2. Définition de l'ADN:

L'acide désoxyribonucléique (ADN) est une molécule géante qui contient l'information génétique de tous les organismes vivants. Il est composé de deux brins enroulés en forme d'hélice, appelés chromosomes.

Chaque brin est composé d'unités de base appelées nucléotides. Chaque nucléotide est composé de trois composants:

- ***Base azotée:*** Il existe quatre types de bases azotées dans l'ADN: adénine (A), guanine (G), thymine (T) et cytosine (C).
- ***Sucre à cinq carbones***: soit ribose (dans l'ADN ribosique) soit désoxyribose (dans l'ADN désoxyribosique).

- ***Groupe phosphate***: relie les nucléotides les uns aux autres.

La séquence des bases azotées le long de la chaîne d'ADN est ce qui code 'information génétique. Cette information est lue par les cellules et utilisée pour former des protéines, qui sont des molécules essentielles à toutes les fonctions du corps.

3. Représentation de la structure de l'ADN:

La structure de l'ADN peut être représentée de différentes manières, notamment:

- **Modèle de l'échelle en spirale:** C'est le modèle le plus courant, où les deux brins d'ADN sont enroulés comme les barreaux d'une échelle, avec les bases azotées appariées sur les côtés de l'échelle.

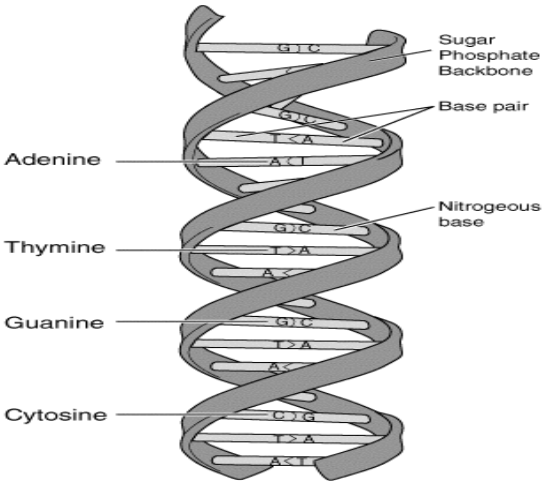


***Figure** 4.1 : Représentation graphique des principales caractéristiques d'ADN.* [54]

4. **Méthodologie :** Le système proposé est un nouvel algorithme de cryptage basé sur le motif de l'ADN. Il y a trois étapes dans cet algorithme - le cryptage, la génération de table de clés aléatoires et le décryptage. Dans la première étape, les données source sont cryptées en utilisant un codage binaire et des règles complémentaires sont appliquées [52]. Dans la deuxième étape, une clé aléatoire est générée et utilisée pour le prochain niveau de cryptage. Dans la troisième étape, le processus de décryptage a lieu, ce qui est le processus inverse du cryptage. La figure 1 donne le diagramme de flux du système proposé.

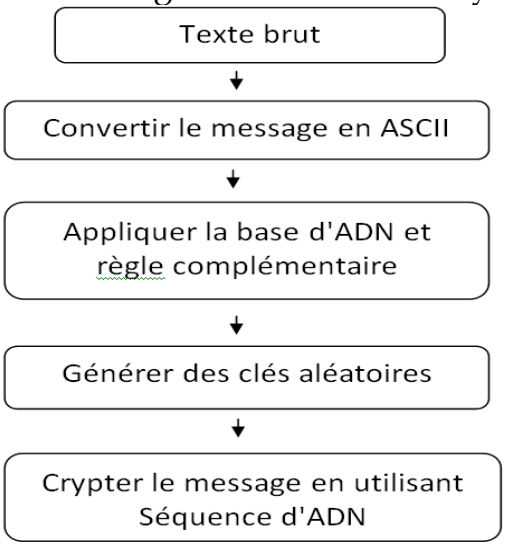


***Figure 4.2.** Organigramme de la méthode proposée* [55]

La technique de masquage basée sur l'ADN utilise les quatre nucléotides A, C, G et T en séquence pour former une séquence d'ADN. La base de données EBI de l'Institut européen de bioinformatique est utilisée pour extraire la séquence d'ADN. Litmus se compose de 154 nucléotides d'ADN et Balsaminaceae se compose de 2283 nucléotides d'ADN. La représentation de Litmus est donnée ci-dessous[49].

Litmus

"ATCGAATTCGCGCTGAGTCACAATTCGCGCTGAGTCACAATTCGCGCTGAGTCACAATTGTG ACTCAGCCGCGAATTCCTGCAGCCCCGAATTCCGCATTGCAGAGATAATTGTATTTAAGTGC CTGCTCGATACAATAAACGCCATTTGACC" [50].

4.1 Algorithme de chiffrement

Tout d'abord, l'expéditeur choisit un message original M qui doit être envoyé au destinataire via un réseau. Il y a donc différentes sous-étapes pour fournir la forme finale du message et l'envoyer au destinataire. Le processus de chiffrement se déroule selon les étapes suivantes.

Étape *1* : Considérez un message M.
Étape *2* : Convertissez le message en une forme ASCII.
Étape *3 :* Convertissez le code ASCII en une forme codée binaire de telle sorte que chaque caractère soit représenté à l'aide de 8 bits.
Étape *4:* Représenter ces données binaires sous forme codée d'ADN selon le tableau 1
Étape *5:* Appliquer la règle complémentaire pour la séquence d'ADN.
Étape *6:* Maintenant, former un groupe de deux nucléotides dans une séquence d'ADN.
Étape *7:* Attribuer la valeur d'index à chaque paire de nucléotides dans la séquence de référence d'ADN. Pour attribuer cette valeur d'index, la valeur binaire de la paire correspondante est convertie en décimal en utilisant le tableau 2.
Étape *8:* Le message crypté est généré.

EXEMPLE

Étape 1 : Considérez un message M.
T U R N
Étape 2 : Convertissez le message en une forme ASCII.
84 85 82 78
Étape 3 : Convertissez le code ASCII en une forme codée en binaire de telle sorte que chaque caractère soit représenté en utilisant 8 bits
01010100 01010101 01010010 01001110
Étape4 :Représentez ces données binaires sous forme codée en ADN selon le tableau1
CCCA CCCC CCAG CATG
Étape 5: Appliquer la règle complémentaire pour la séquence d'ADN.
[(A ➔C) (C ➔ G) (G➔ T) (T ➔A)] GGGC GGGG GGCT GCAT
Étape 6: Maintenant, former un groupe de deux nucléotides dans une séquence d'ADN. GG GC GG GG GG CT GC AT

Étape 7: Attribuer la valeur d'index à chaque paire de nucléotides dans la séquence de référence de l'ADN. Pour attribuer cette valeur d'index, la valeur binaire de la paire correspondante est convertie en décimal en utilisant le tableau 2.
Litmus: "AT_3 CG_6 AA_0 TT_{15} CG_6 CG_6 CT_7 GA_8 GT_{11} CA_4 CA_4 AT_3 TC_{13} GC_9 GC_9 TG_{14} AG_2 TG_{14} AA_0 CC_5 GG_{10}"
Étape 8: Le message crypté est généré. 10 9 10 10 10 7 9 3

4.2 Génération de clé

Ceci est la deuxième étape du système proposé. Plusieurs autres opérations peuvent être effectuées pour mieux masquer les informations, par exemple en examinant quatre types de bases d'acides nucléiques tels que l'adénine (A), la guanine (G), la cytosine (C) et la thymine (T). Un couple de nucléotides est formé et des valeurs binaires sont créées pour un cryptage ultérieur. Des valeurs binaires de quatre bits sont attribuées à chaque paire dans un tableau d'index composé de bases azotées telles que A, T, G et C. Un tableau d'index est créé en fonction de l'ordre des quatre lettres A, G, T et C.

ADN	Bits binaires	ADN	Bits binaires
AA	0000	GA	1000
AC	0001	GC	1001
AG	0010	GG	1010
AT	0011	GT	1011
CA	0100	TA	1100
CC	0101	TC	1101
CG	0110	TG	1110
CT	0111	TT	1111

Tableau 4.2 : *Paire d'ADN avec la valeur binaire correspondante*

4.3 décryptage

Le décryptage est le processus de conversion d'un texte crypté en message original. Il ne peut être effectué que par un utilisateur autorisé qui possède une clé secrète ou un mot de passe. Le destinataire utilise une méthode pour identifier et extraire les informations originales cachées dans la séquence d'ADN. Le message crypté est reçu sous forme de chaînes de nombres et de séquences d'ADN. L'extraction des données commence en comparant la séquence d'ADN et les chaînes de nombres. Ensuite, une règle complémentaire est appliquée à la séquence d'ADN obtenue. Le code de base d'ADN est ensuite converti en valeurs binaires, à partir desquelles les valeurs ASCII sont représentées Enfin, les données originales sont obtenues.

- L'algorithme de cryptage basé sur l'ADN garantit la sécurité des informations en cachant les données d'une manière qui rend difficile pour l'attaquant de les décrypter. Le choix aléatoire des nucléotides de l'ADN assure un transfert sécurisé

des messages à travers le réseau, offrant ainsi une sécurité accrue des informations. Seul le destinataire prévu peut décrypter les données cryptées pour accéder aux données d'origine. Dans ce but, nous proposons un algorithme que nous discuterons ci-dessous pour garantir une sécurité et une confidentialité accrues des données pendant leur transfert.

5 Méthodologie proposée

Le système proposé est un algorithme de cryptage complexe basé sur le modèle de l'ADN. Il y a six étapes dans cet algorithme "algorithme de cryptage" –
1- Crypter les données en utilisant la clé(données *XOR* clé) et créer une chaîne encodée d'ADN
-2 Créer un ensemble de chromosomes. 3- Choisir le type de croisement en fonction d'une valeur aléatoire entre 0 et 1. *(Soit le croisement croisé est choisi. Ou choisir un échange de point unique sur les individus. Sinon, les deux types de croisement génétique sont appliqués successivement.)* -4 Chaque chromosome est entièrement tourné entre le point **1** et le point **2**. -5 Changer aléatoirement les règles de l'ADN en désoxyribonucléique d'une autre. 6- Appliquer la transformation en utilisant "l'addition" et "modifier les règles de l'ADN" en dernier.

Une clé aléatoire est créée et utilisée pour le niveau suivant de cryptage avec sa mise à jour. À chaque étape, on parvient enfin à la chaîne finale encodée d'ADN, schéma du système proposé.

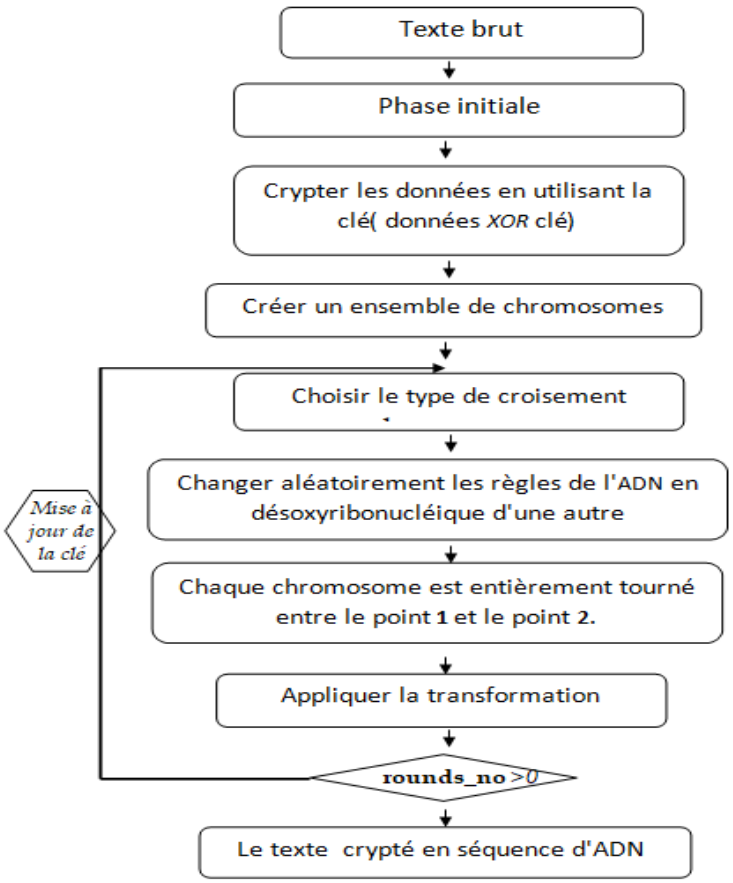


Figure 4.3. Organigramme de la méthode proposée [55]

1. **Phase initiale :** à cette Phase, les choses se passent comme suit
   - ***Déterminer le nombre de répétitions ou de tours*.**

Le nombre de tours est généré de manière aléatoire. Le résultat est attribué à une variable appelée "*rounds_no*". Un nombre entier aléatoire est généré entre **1** et **5** (*inclus*) avec un pas de **2**. Ce nombre peut être utilisé pour déterminer le nombre de répétitions ou de tours dans les opérations. (*etapes*).

   - ***Génération de clé aléatoire :***

(peut être de n'importe quelle longueur, peut être la longueur du texte original)

Dans ce cas, une clé de 128 bits a été utilisée.

2. Crypter les données en utilisant la clé(données *XOR* clé) :

(Répétez la clé ***UNIQUEMENT*** *si les données sont plus longues que la clé et chiffrez)*
La longueur des données binarisme et de la clé est vérifiée avant le chiffrement. Si les données sont plus longues que la clé, la clé est étendue pour correspondre à la longueur des données. Ensuite, une opération XOR est effectuée bit à bit entre les données et la clé étendue pour chiffrer les données et convertir en séquence d'**ADN** chiffrée

3. Créer un ensemble de chromosomes :

Toutes les combinaisons possibles de la longueur de la chaîne d'ADN sont calculées et un nombre aléatoire de chromosomes est choisi dans la population. Ensuite, la longueur de la chaîne d'ADN est divisée par le nombre de chromosomes pour obtenir la longueur de chaque chromosome. Ensuite, les chromosomes sont créés et la clé est mise à jour .

4. Choisir le type de croisement :

le type de croisement est choisi en fonction d'une valeur aléatoire entre **0** et **1**. Si la valeur est inférieure à **0,33**, le croisement de rotation est choisi.

- Dans ce cas : Une valeur est choisie au hasard entre **1** et la longueur du chromosome, et un nombre aléatoire entre **0** et 1 est généré pour choisir la rotation soit vers la droite, soit vers la gauche. Si le nombre aléatoire est supérieur à **0,5**, une rotation cyclique vers le droit est effectuée sur le chromosome, et la partie déplacée est fusionnée avec le début d'origine, puis le chromosome résultant est ajouté à la liste de la nouvelle population. Si le nombre aléatoire est inférieur ou égal à **0,5**, une rotation cyclique vers la gauche est effectuée et la partie restante est fusionnée avec la partie déplacée, puis le chromosome résultant est ajouté à la liste de la nouvelle population.

Ce processus est répété pour chaque chromosome dans la population actuelle.

- Si la valeur est entre **0,33** et **0,66**, les paires de chromosomes sont déterminées dans la population. Le premier chromosome est attribué au candidat **1** et le

deuxième chromosome au candidat **2**. Ensuite, un point de croisement aléatoire est choisi entre **0** et la longueur du chromosome. La première descendance est créée en fusionnant la première partie du candidat **2** avec la partie restante du candidat **1**, et la deuxième descendance est créée en fusionnant la première partie du candidat **1** avec la partie restante du candidat **2**. Les descendants sont ajoutés à la nouvelle population. Ensuite

. Sinon, les deux types d'échanges génétiques sont appliqués séquentiellement.

5. **Changer aléatoirement les règles de l'ADN en désoxyribonucléique d'une autre :**

Un dictionnaire a été créé pour le processus de retournement des nucléotides intégré en exécutant la boucle deux fois pour créer une paire de bases. Une base est choisie au hasard dans une liste de bases et assignée à base**1**, puis une autre base est choisie de la même manière et assignée à base**2**. Une paire clé-valeur est ajoutée au dictionnaire où la clé est base**1** et la valeur est base**2**, et la même opération est effectuée pour une autre paire. Cela garantit que chaque paire de bases est inversée de manière cohérente lors du processus de modification de la séquence d'ADN

6. **Chaque chromosome est entièrement tourné entre le point 1 et le point 2 :**

Ce processus sélectionne une tranche de chromosome définie par deux points (point **1** et point **2**).Il inverse les bits (complément) à l'intérieur de la tranche sélectionnée. Dans ce cas, **'0'** devient **'1'** et vice versa.

Les éléments en dehors de la plage sélectionnée restent inchangés.

7. **Application de la mutation en utilisant ''complément'' et ''modification des règles de l'acide nucléique'' :**

Ce processus est répété pour chaque chromosome dans le groupe actuel. Le chromosome d'ADN est converti en une séquence binaire en utilisant des espaces et des clés de déchiffrement. Un point de départ et un point de fin aléatoires sont choisis pour muter la séquence de bits. Des informations sur les points de mutation sélectionnés sont ajoutées à la clé de déchiffrement. Le chromosome binaire est converti en groupes de **4** bits, et si le dernier élément est incomplet, les deux derniers bits sont extraits et convertis en une base d'ADN en utilisant une table de conversion prédéfinie. Les quatre groupes de bits sont convertis en une séquence d'ADN en utilisant la fonction bits_to_dna et la table de conversion. Si un groupe est incomplet à la fin, la base d'ADN extraite est ajoutée à la fin de la séquence d'ADN

5.2 Algorithme de chiffrement Proposé

Tout d'abord, le texte original "a" est sélectionné par l'expéditeur pour être envoyé au Cloud via le réseau. Ensuite, des étapes secondaires différentes sont entreprises pour atteindre la forme finale du texte et l'envoyer au Cloud. Le processus de cryptage est effectué selon les étapes suivantes.

- **Étape *1*** : Considérez un texte T.
- **Étape *2*** : Déterminer le nombre de répétitions ou de tours**.**
- **Étape *3*** : binariser les données et les convertir en séquence d'ADN.
- **Étape *4 :*** Chiffrer les données avec la clé : données XOR clé.

- Représenter ces données binaires sous forme codée d'ADN selon le tableau 1

- **Étape *5:*** Générer une population de chromosomes
- **Étape *6:*** Choisir le type de croisement
- **Étape *7:*** Changer aléatoirement les règles de l'ADN d'une autre
- **Étape *8:*** Faites tourner chaque chromosome complet entre le point 1 et le point 2..
- **Étape *9 :*** Nous appliquons la transformation en utilisant les techniques de "complément" et de "modification des règles de l'acide nucléique".
- **Étape *10***: Le texte crypté est généré

EXEMPLE

Étape 1 : Considérez un texte M. '' a ''

Étape 2 : Déterminer le nombre de répétitions ou de tours**. +** Génération de clé aléatoire *(Dans ce cas, le nombre de tours a été fixé à un seul.)*

Clé : 00110111011001100111100001001001010010010100010001101010010010010100111101100110011011010110001101110000010010100011001001111000

Étape 3 : binariser les données et les convertir en séquence d'ADN.

['01', '10', '00', '01'] ⟶ CGAC

Étape 4 : Chiffrer les données avec la clé : données XOR clé.

Les données sont converties en format binaire.	*0*	*1*	*1*	*0*	*0*	*0*	*0*	*1*	
Séquence initiale de l'ADN	**C**		**G**		**A**		**C**		
Clé	*0*	*0*	*1*	*1*	*0*	*1*	*1*	*1*	….
Chiffrer les données avec la clé (XOR clé)	*0*	*1*	*0*	*1*	*0*	*C*	*1*	*0*	
les convertir Données cryptées en séquence d'ADN	*C*		*C*		*A*		*G*		

Données cryptées en séquence d'ADN : CCAG

- ***Étape 5:*** Générer une population de chromosomes

  La longueur de la chaîne d'ADN est divisée par le nombre de chromosomes pour obtenir la longueur de chaque chromosome. Ensuite, les chromosomes sont créés et la clé est mise à jour

  et la population des chromosomes est créée. *(sachons que le nombre des chromosomes 2)*

  - *la population des chromosomes* : ['CC', 'AG']
  - *Mise à jour de la clé*

  <key>0011011101100110011110000100100101001001010001000110101001001001010011110110011001101101011000110111000001001010001100100111100011001100110110101100011011100000100101000110010011110001100110011011010110001101110000010010100011001001111000<key><no_rounds>1<no_rounds><round><reshape>2<reshape>

- ***Étape 6:*** Choisir le type de croisement

Dans cet exemple, la méthode de croisement de rotation a été choisie (fusion de chaque paire de chromosomes dans la population en utilisant le processus de reproduction sexuée à un point donné). Étant donné que la valeur aléatoire obtenue est comprise entre 0,33 et 0,66, les paires de chromosomes ont été déterminées dans la population. Le premier chromosome **['CC']** a été attribué au candidat 1 et le deuxième chromosome **['CG']** au candidat 2. Ensuite, un point de croisement a été choisi, qui est 0. La première descendance a été créée en fusionnant la première partie du candidat 2 **[' ']** avec la partie restante du candidat 1 **['CC']**, et la deuxième descendance a été créée en fusionnant la première partie du candidat 1 **[' ']** avec la partie restante du candidat 2 **['CG'].** Ensuite, les descendants ont été ajoutés à la nouvelle population.

- La population après l'application de le croisement. ['CC', 'CG']
- *Mise à jour de la clé*

<key>0011011101100110011110000100100101001001010001000110101001001001010011110110011001101101011000110111000000100101000110010011111000<key><no_rounds>1<no_rounds><round><reshape>2<reshape><crossover><type>single_point_crossover<type><single_point>0|<single_point><crossover>

- ***Étape 7:*** Changer aléatoirement les règles de l'ADN d'une autre
  - {'A'→ 'C', 'C'→'A', 'T'→ 'G', 'G'→ 'T'}
  - *Mise à jour de la clé*

<key>0011011101100110011110000100100101001001010001000110101001001001010011110110011001101101011000110111000000100101000110010011111000<key><no_rounds>1<no_rounds><round><reshape>2<reshape><crossover><type>single_point_crossover<type><single_point>0|<single_point><crossover><mutation><mutation_table>{'A':'C','C':'A','T':'G','G':'T'}<mutation_table

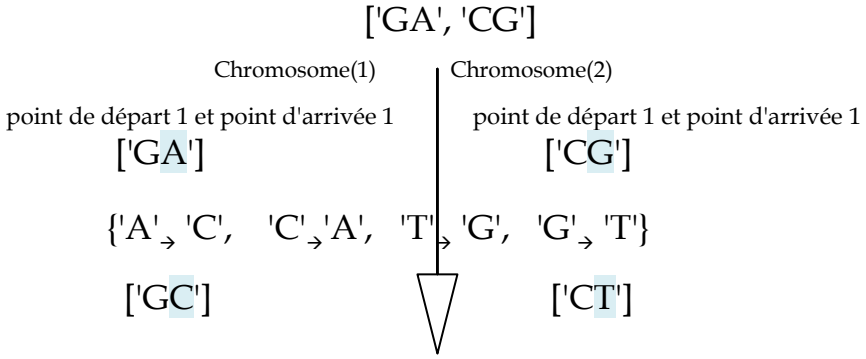
- ***Étape 8:*** Faites tourner chaque chromosome complet entre le point 1 et le point 2. À partir du premier chromosome (0101), nous déterminons la tranche entre les points (point 3 et point 3) ['0101']. Nous inverserons les bits (complément) à l'intérieur de la tranche sélectionnée. Dans ce cas, '0' devient '1' et vice versa, tandis que les éléments en dehors de la plage sélectionnée restent dans leur état d'origine. ⟶ ['0100'] CA de la deuxième chromosome [' 0110'], nous déterminons l'intervalle entre les points (point 0 et point 1). Nous inverser les bits. ⟶ [' 1010'] GG
- ***Étape 9:*** Nous appliquons la transformation en utilisant :
  - ***la techniques de "complément" :*** Selon le tableau suivant.

0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
TA	***TC***	***TG***	***TT***	***GA***	***GC***	***GG***	***GT***	***CA***	***CC***	***CG***	***CT***	***AA***	***AC***	***AG***	***AT***

tableau 4.3 tableau standard pour rechercher des paires de bases nucléiques

['CA', 'GG'] ⟶ ['0100'] [' 1010'] —*complément*→ ['GA', 'CG']

- ***et de "modification des règles de l'acide nucléique***" : Nous choisissons un point de départ et un point d'arrivée pour convertir la séquence pour chaque chromosome.



- ***Étape 10***: Le texte crypté est généré. GCCT

5.3 Génération de clé

La clé a été mise à jour à chaque étape précédente pour améliorer la dissimulation des informations de manière plus efficace.

<key>0011011101100110011110000100100101001001010001000110101001001001010011111011001100110110101100011011100000100101000110010011110000<key><no_rounds>1<no_rounds><round><reshape>2<reshape><crossover><type>single_point_crossover<type><single_point>0|<single_point><crossover><mutation><mutation_table>{'A':'C','C':'A','T':'G','G':'T'}<mutation_table><chromosome><complement_mutation>(3,3)<complement_mutation><alter_mutation>(1,1)<alter_mutation><chromosome><chromosome><complement_mutation>(0,1)<complement_mutation><alter_mutation>(1,1)<alter_mutation><chromosome>

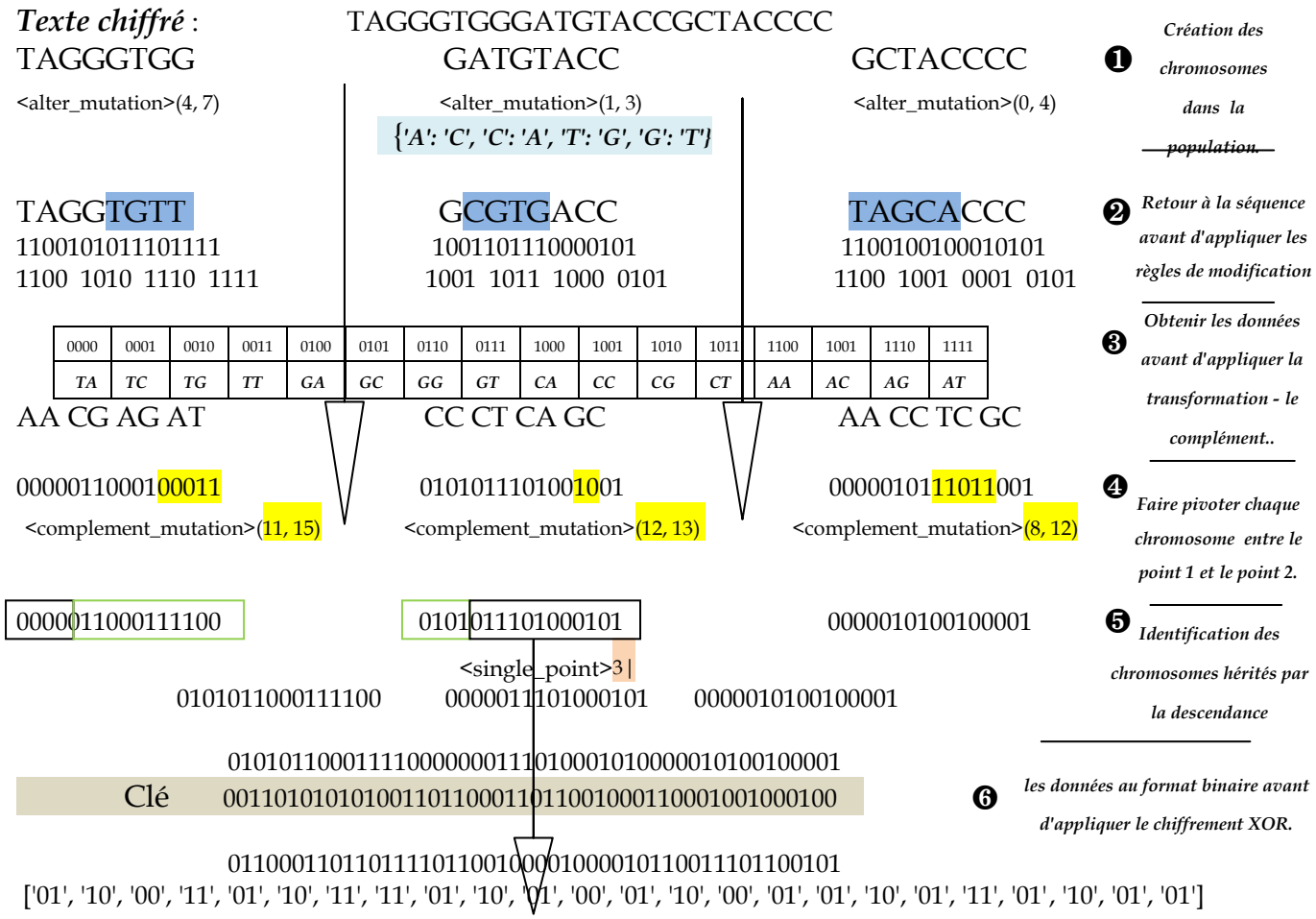
- Dans l'exemple suivant, nous expliquerons comment décrypter le texte en utilisant la clé fournie.
  - ***clé de déchiffrement***

0011010101010011011000110110010001100010010001000100111101100001010011010111010001100111010000110101001100110001011100000110000<key><no_rounds>1<no_rounds><round><reshape>8<reshape><crossover><type>single_point_crossover<type><single_point>3|<single_point><crossover><mutation><mutation_table>{'A': 'C', 'C': 'A', 'T': 'G', 'G': 'T'}<mutation_table><chromosome><complement_mutation> (11, 15)<complement_mutation><alter_mutation>(4, 7)<alter_mutation><chromosome><chromosome> <complement_mutation>(12, 13)<complement_mutation><alter_mutation>(1, 3)<alter_mutation> <chromosome><chromosome><complement_mutation>(8, 12)<complement_mutation><alter_mutation> (0, 4)<alter_mutation><chromosome>

A. Interprétation de la clé de déchiffrement:

①: le Clé ② : le nombre de répétitions ou de tours. ③: la longueur de chromosome ④ : Règles de l'acide désoxyribonucléique modifié ⑤ : Points de croisement chromo-somique ⑥ : Le début et la fin de la séquence qui sera transcrit dans chaque chromosome. ⑦ : Le point de départ et le point d'arrêt de la conversion de séquence pour chaque chromosome.

B. *Explication des étapes*



Text original :

['01', '10', '00', '11'] ['01', '10', '11', '11'] ['01', '10', '01', '00'] ['01', '10', '00', '01'] ['01', '10', '01', '11'] ['01', '10', '01', '01']

C O D A G E

- ***Text original : CODAGE***

6. Résultats expérimentaux

Les procédures proposées sont mises en œuvre sur la plateforme Python. L'algorithme de chiffrement est mis en œuvre avec succès avec un texte en clair d'entrée de taille spécifiée. Des expériences ont été menées pour le chiffrement et le déchiffrement du document texte.
Le texte chiffré peut être obtenu en peu de temps. La capture d'écran du processus de chiffrement est représentée ci-dessous.

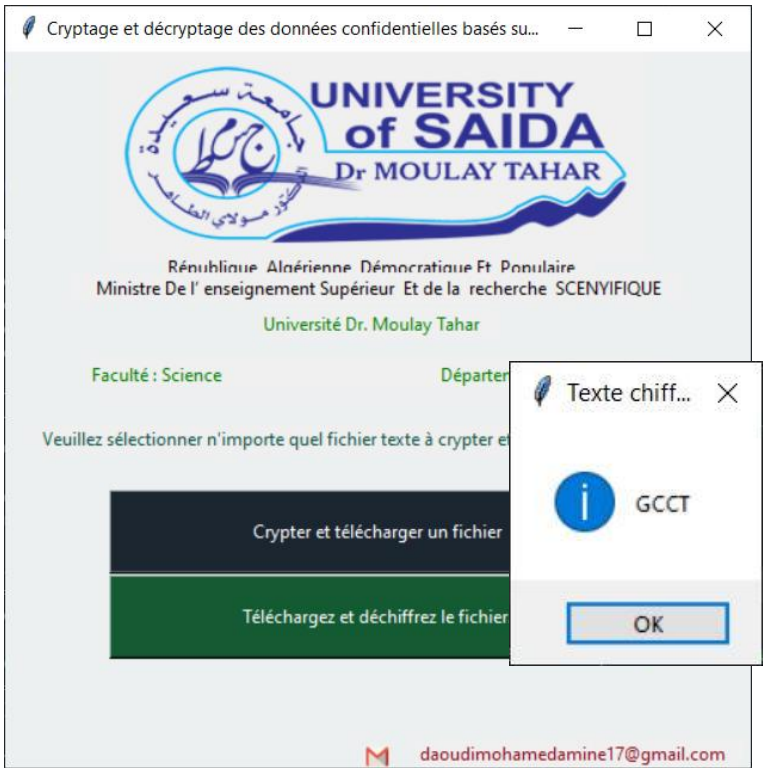


Figure 4.4: *Cryptage de texte* [55]

7. Conclusion

Le chiffrement des données avant de les envoyer vers le Cloud est une étape cruciale pour protéger les informations sensibles. Par conséquent, lors du choix de l'algorithme de chiffrement, il est nécessaire d'évaluer celui-ci en fonction de modèles de données variés, de mesurer la vitesse de chiffrement et de déchiffrement, ainsi que la consommation des ressources. Il est également important d'évaluer soigneusement les risques et les avantages avant de prendre toute décision.

Chapitre 5

Implémentation et Évaluation

1. Introduction

Après avoir créé l'algorithme et défini notre approche pour sécuriser les données avant de les envoyer vers le Cloud Computing dans le chapitre précédent, nous passerons à la phase de mise en œuvre et d'évaluation de notre stratégie de chiffrement des données. Ainsi, dans ce chapitre, nous présenterons l'environnement de développement et d'exécution, les résultats obtenus, ainsi que les résultats expérimentaux.

2. langues et environnement de développement

Nous avons développé l'algorithme sur une machine équipé d'un processeur Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz, avec **8** Go de *RAM*, sous Windows **8 64** bits. Nous avons utilisé l'environnement de développement ***Spyder***

2.1. Le langage de programmation Python

Python est un langage de programmation interprété, orienté objet et de haut niveau, créé par Guido van Rossum et publié pour la première fois en 1991. Parmi ses caractéristiques clés, on peut citer :

- ***Syntaxe simple et lisible*** : La syntaxe de Python est conçue pour être simple et facile à lire, ce qui en fait un excellent choix pour les débutants en programmation.
- Langage multi-paradigme : Python prend en charge plusieurs paradigmes de programmation, y compris la programmation impérative, fonctionnelle et orientée objet.
- ***Vaste bibliothèque standard*** : Python dispose d'une vaste bibliothèque standard qui offre des fonctionnalités pour une variété de tâches, telles que la manipulation de fichiers, le traitement de données, la création d'interfaces graphiques utilisateur, et bien plus encore.
- ***Portabilité*** : Python est un langage de programmation portable, ce qui signifie qu'il peut être exécuté sur différents systèmes d'exploitation, y compris Windows, macOS et Linux.
- ***Large communauté*** : Python bénéficie d'une grande communauté de développeurs qui contribuent à son écosystème en développant des bibliothèques tierces, en fournissant des ressources d'apprentissage et en offrant un support technique.
- Utilisation dans divers domaines : Python est largement utilisé dans une variété de domaines, y compris le développement web, l'analyse de données, l'intelligence artificielle, l'apprentissage automatique, l'automatisation des tâches système, et bien d'autres.

Le langage de programmation Python est devenu l'un des langages les plus célèbres et largement utilisés dans le monde, grâce à sa facilité d'utilisation et sa polyvalence.

2.2 Environnements de développement spyder

Spyder est un environnement de développement intégré (IDE) spécialement conçu pour les scientifiques et les ingénieurs qui utilisent Python pour le calcul numérique, l'analyse de données et d'autres tâches liées aux sciences. Voici quelques-unes des caractéristiques principales de Spyder :

- ***Éditeur de code avancé*** : Spyder dispose d'un éditeur de code puissant avec des fonctionnalités telles que la coloration syntaxique, l'indentation automatique, la complétion automatique et la mise en surbrillance des erreurs.
- ***Explorateur de variables*** : Il offre un explorateur de variables intégré qui permet d'afficher et d'explorer les variables actuellement définies dans l'environnement de travail.
- ***Console IPython intégrée*** : Spyder intègre une console IPython interactive qui permet d'exécuter du code Python en temps réel et d'explorer les résultats de manière interactive.
- ***Outils d'analyse de données :*** Il comprend plusieurs outils d'analyse de données, tels que des tableaux de données interactifs (DataFrame) basés sur la bibliothèque pandas, des graphiques matplotlib intégrés et des outils de profilage de code.
- ***Intégration avec des outils scientifiques*** : Spyder est intégré à plusieurs bibliothèques et outils scientifiques populaires tels que NumPy, SciPy, matplotlib, SymPy et bien d'autres, ce qui en fait un choix idéal pour les tâches scientifiques et d'ingénierie.
- ***Personnalisable et extensible*** : Il est possible de personnaliser et d'étendre Spyder grâce à son architecture modulaire et à son support pour les plugins.
- ***Interface utilisateur conviviale :*** Spyder propose une interface utilisateur conviviale avec une disposition intuitive des fenêtres et des onglets, ce qui facilite la navigation et l'utilisation de l'IDE.

Spyder est un environnement de développement intégré puissant et convivial, spécialement conçu pour les scientifiques et les ingénieurs travaillant avec Python dans le domaine du calcul numérique et de l'analyse de donnée

3. Évaluation des performances de l'algorithme de chiffrement:

- **Facteurs d'évaluation:**

L'évaluation des performances de l'algorithme de chiffrement repose sur plusieurs facteurs, dont voici quelques-uns:

3.1 Vitesse de chiffrement:

L'algorithme de chiffrement doit être rapide pour chiffrer et déchiffrer les textes, tout en respectant les exigences de sécurité.
La vitesse de chiffrement peut être mesurée en calculant le temps nécessaire pour chiffrer et déchiffrer les textes, en étudiant un échantillon de textes de différentes tailles

- **Le temps nécessaire pour crypter un texte :**

Pour évaluer les performances, nous avons calculé le temps nécessaire pour coder un ensemble de textes de différentes tailles. Dans cette expérience, un échantillon de **20** textes de tailles différentes a été utilisé, chaque texte étant codé en **3** tentatives avec des clés de codage différentes.

(*Dans notre expérience, nous avons créé des chaînes aléatoires de lettres, de chiffres, de symboles de ponctuation et d'espaces avec un certain nombre en utilisant le random en Python.)*

Les résultats sont présentés dans le tableau suivant.

Le nombre de caractères dans le texte		Le temps total nécessaire pour crypter le texte (en secondes)			
		Test 01	Test 02	Test 03	Le temps moyen
Texte **(01)**	1000	0.0897	0.0478	0.0837	0.0737
Texte **(02)**	2000	0.1436	0.0548	0.0927	0.0970
Texte **(03)**	3000	0.9254	0.0817	0.2234	0.4101
Texte **(04)**	4000	0.1078	0.1087	0.1726	0.1297
Texte **(05)**	5000	0.1366	0.1609	0.1629	0.1534
Texte **(06)**	6000	0.2273	0.1585	0.6442	0.3433
Texte **(07)**	7000	0.1834	0.1994	0.1804	O.1877
Texte **(08)**	8000	0.2184	0.3291	1.7845	0.7773
Texte **(09)**	9000	0.2593	0.2513	0.2363	0.2489
Texte **(10)**	10000	0.3046	0.3261	0.2632	0.2979
Texte **(11)**	11000	6.7678	0.2983	0.3371	2.4677
Texte **(12)**	12000	0.3191	1.720	0.3919	0.8103
Texte **(13)**	13000	0.3451	6.7626	0.3574	2.4883
Texte **(14)**	14000	0.3890	0.3889	1.3426	0.7068
Texte **(15)**	15000	1.2296	0.4102	0.5558	0.7318
Texte **(16)**	16000	0.7180	0.4454	0.4459	0.5364
Texte **(17)**	17000	0.4717	0.5194	0.4986	0.4965
Texte **(18)**	18000	0.4936	0.5187	0.4806	0.4976
Texte **(19)**	19000	7.1373	0.9885	0.5305	2.8854
Texte **(20)**	20000	0.5692	0.6788	1.2454	0.8311

***Tableau 5.1:** Le temps total nécessaire pour crypter Un texte*

20 textes de différentes tailles ont été utilisés, allant de 1000 à 20000 caractères. Cette gamme de tailles de données est considérée comme bonne pour évaluer les performances de l'algorithme.

Plateforme informatique:

Un ordinateur fonctionnant sous Windows 8 64 bits avec un processeur Intel Core i5-8250U et 8 Go de RAM a été utilisé.

Ces spécifications sont de niveau moyen, ce qui les rend généralisables à un large éventail de dispositifs.

Le temps total d'encodage de chaque texte a été mesuré avec trois tentatives différentes en utilisant des clés de codage différentes.

Cette approche est également bonne pour évaluer les performances moyennes de l'algorithme.

Les résultats ont montré que le temps d'exécution augmente presque linéairement avec la taille des données. Ce comportement est attendu car les algorithmes ont généralement besoin de plus de temps pour traiter de plus grandes quantités de données.

Dans l'ensemble, le temps d'exécution était acceptable. Le temps d'encodage le plus rapide pour un texte de 1000 caractères était de 0,0478 seconde, et le temps le plus lent pour un texte de 20000 caractères était de 1,7845 seconde. Ces chiffres peuvent être comparés à d'autres algorithmes de chiffrement traditionnels.

Il y avait quelques différences de temps d'exécution entre les trois tests pour chaque texte. Cela peut être dû à des facteurs différents tels que la vitesse du processeur de l'ordinateur, les processus en arrière-plan en cours d'exécution et de légères variations dans les données d'entrée.

La consommation d'énergie est un facteur important dans l'évaluation de la faisabilité du chiffrement a base de l'ADN, et en suivant les variations du niveau de la batterie pendant l'expérience, il a été observé que le taux de consommation était relativement faible.

Dans l'ensemble, les résultats de l'expérience montrent que l'algorithme de chiffrement utilisant l'ADN peut atteindre un temps d'exécution acceptable.

Cependant, davantage de recherches sont nécessaires pour évaluer la consommation d'énergie de cet algorithme et déterminer ses applications pratiques**.**

- **Le temps nécessaire pour crypter un texte :**

*Nous avons calculé le temps nécessaire pour décrypter un ensemble de textes en trois tentatives différentes en utilisant la même clé de décryptage. Les résultats sont indiqués dans le tableau ci-**dessous.***

Le nombre de caractères dans le texte		Le temps total nécessaire pour crypter le texte (en secondes)				Le temps total nécessaire pour décrypter (en secondes)			
		Test 01	Test 02	Test 03	Le temps moyen	Test 01	Test 02	Test 03	Le temps moyen
Texte **(01)**	1000	0.0897	0.0478	0.0837	0.0737	0.0813	0.0759	0.0791	0.0827
Texte **(02)**	1100	0.0349	0.0309	0.0438	0.0365	0.0787	0.0850	0.0825	0.0820
Texte **(03)**	1200	0.0598	0.1056	0.0319	0.0657	0.0438	0.0676	0.0570	0.0561
Texte **(04)**	1300	0.0538	0.0561	0.0365	0.0488	0.0507	0.0564	0.0425	0.0519
Texte **(05)**	1400	0.0410	0.0985	0.0378	0.0591	0.0752	0.0598	0.0643	0.0664
Texte **(06)**	1500	0.0608	0.0489	0.2689	0.1262	0.1461	0.1754	0.1799	0.1671
Texte **(07)**	1600	0.0477	0.0817	0.0675	0.0656	0.107	0.0953	0.0888	0.0970
Texte **(08)**	1700	0.0538	0.0615	0.163	0.0927	0.1541	0.125	0.127	0.1353
Texte **(09)**	1800	0.0508	0.0774	0.0747	0.0676	0.0967	0.0910	0.0995	0.0957
Texte **(10)**	1900	0.0588	0.121	0.0538	0.0778	0.0724	0.0794	0.1068	0.0862
Texte **(11)**	2000	0.1436	0.0548	0.0927	0.0970	0.1155	0.1086	0.1136	0.1125

Tableau 5.3: Le temps total nécessaire pour crypter / décrypter Un texte

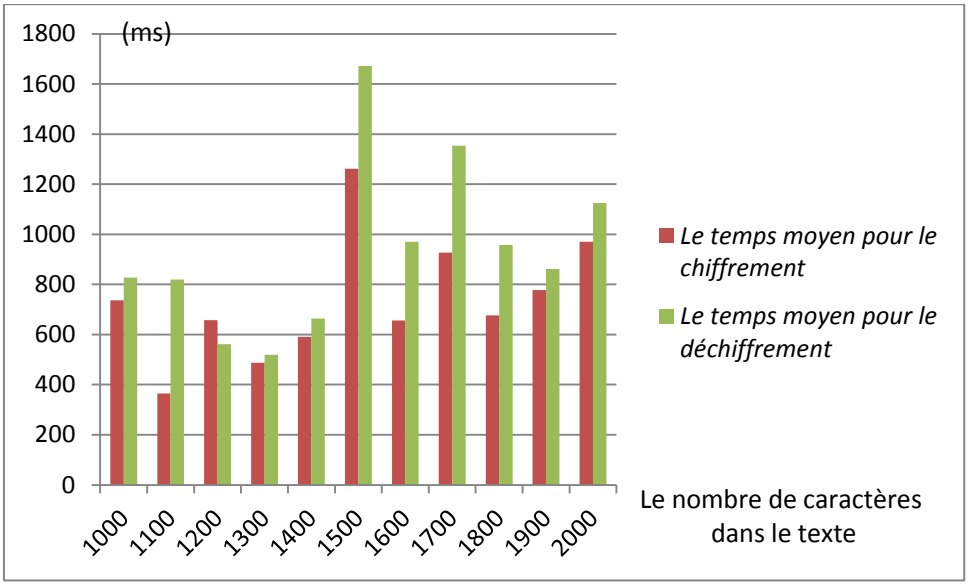


Figure 5.1: Le temps total nécessaire pour crypter / décrypter Un texte [55]

Malgré l'utilisation de la même clé de chiffrement, il a été observé que le temps requis pour décrypter le texte original varie d'une tentative à l'autre en raison de divers facteurs

Il est également remarqué que le temps nécessaire pour crypter le texte est inférieur au temps nécessaire pour le décrypter. Il y a plusieurs facteurs qui ont conduit à une différence dans le temps nécessaire pour chiffrer le texte et le temps nécessaire pour le déchiffrer, malgré l'utilisation de la même clé. Parmi ces facteurs :

- ***La taille du texte*** : Il y a une différence de taille entre le texte chiffré et le texte d'origine, ce qui nécessite un temps supplémentaire pour traiter les données supplémentaires lors du déchiffrement.
- ***L'algorithme utilisé*** : Il y a des différences dans la complexité et l'efficacité des algorithmes utilisés pour le chiffrement et le déchiffrement.
- ***Les performances du matériel et des logiciels*** : Les performances de l'appareil utilisé peuvent avoir un impact sur le temps nécessaire pour le chiffrement et le déchiffrement.
- Les performances peuvent être affectées par la concurrence pour les ressources par d'autres processus utilisant les ressources du système en même temps..

On peut dire que le temps moyen nécessaire pour décrypter l'échantillon complet, estimé à **0.1185** seconde par texte, est considéré comme bon par rapport à la taille relativement grande de la clé de chiffrement. Cette clé garantit la sécurité des données de manière plus efficace.

3.2 Efficacité de l'utilisation de la mémoire:

L'algorithme de chiffrement doit consommer une quantité minimale de mémoire pour fonctionner de manière efficace, en particulier sur les appareils aux ressources limitées.
L'efficacité de l'utilisation de la mémoire peut être mesurée en calculant le rapport entre la quantité de mémoire consommée par l'algorithme et la taille du texte d'origine, comme indiqué ci-dessous.

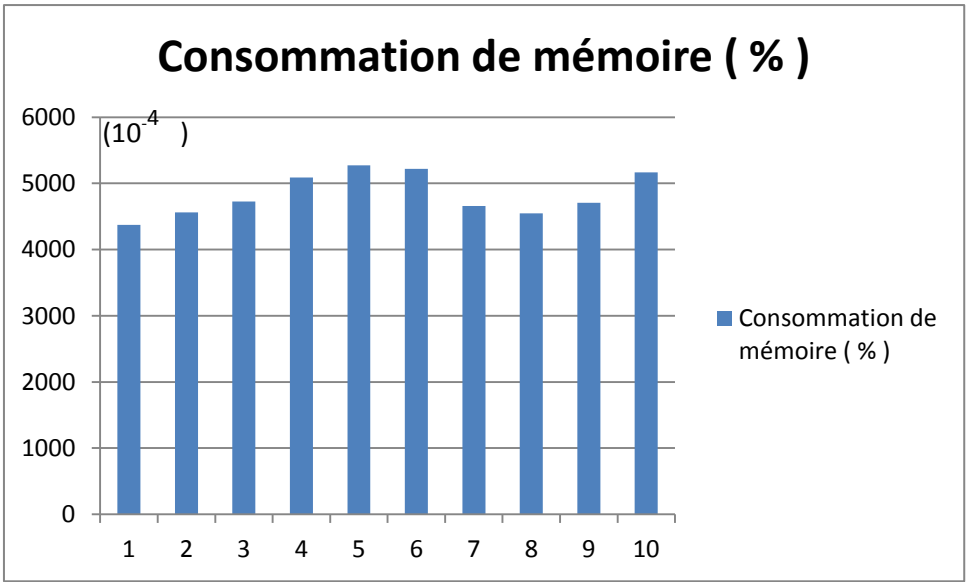


Figure 5.2: *Consommation de mémoire (%)* **[55]**

Le nombre de caractères dans le texte		Le temps total nécessaire pour crypter le texte (en secondes)	Consommation de mémoire (%)
Texte (01)	1000	0.0099	0.4375
Texte (02)	2000	0.068	0.4564
Texte (03)	3000	0.35	0.4727
Texte (04)	4000	1.63	0.5087
Texte (05)	5000	10.48	0.5271
Texte (06)	6000	39.92	0.5220
Texte (07)	7000	0.38	0.4658
Texte (08)	8000	0.16	0.4546
Texte (09)	9000	0.44	0.4706
Texte (10)	10000	46.24	0.5168

Tableau 5.3: Consommation de mémoire (%)

Le tableau montre la relation entre le nombre de caractères dans le texte et la consommation de mémoire (%) pour le cryptage en utilisant un algorithme spécifique.

Remarques:
- En général, plus le nombre de caractères dans le texte est élevé, plus le temps de cryptage et la consommation de mémoire augmentent.
- Il y a des exceptions à cette règle. Par exemple, le cryptage du texte numéro 7 (7000 caractères) ne prend que 0,38 seconde, tandis que le cryptage du texte numéro 9 (9000 caractères) prend 0,44 seconde.
- La raison en est peut-être que le texte numéro 7 contient un motif plus répété que le texte numéro 9, ce qui facilite le cryptage.
- En général, la relation entre le nombre de caractères et le temps de cryptage et la consommation de mémoire est une relation linéaire directe.
- L'algorithme consomme une petite quantité de mémoire pour crypter de petits textes, tandis que la quantité de mémoire consommée augmente considérablement avec de grands textes.

La consommation de mémoire par caractère peut être calculée en divisant la consommation totale de mémoire par le nombre de caractères.
- Dans cet exemple, la consommation de mémoire par caractère varie entre 0,0004375% et 0,0005168%.

Conclusions:

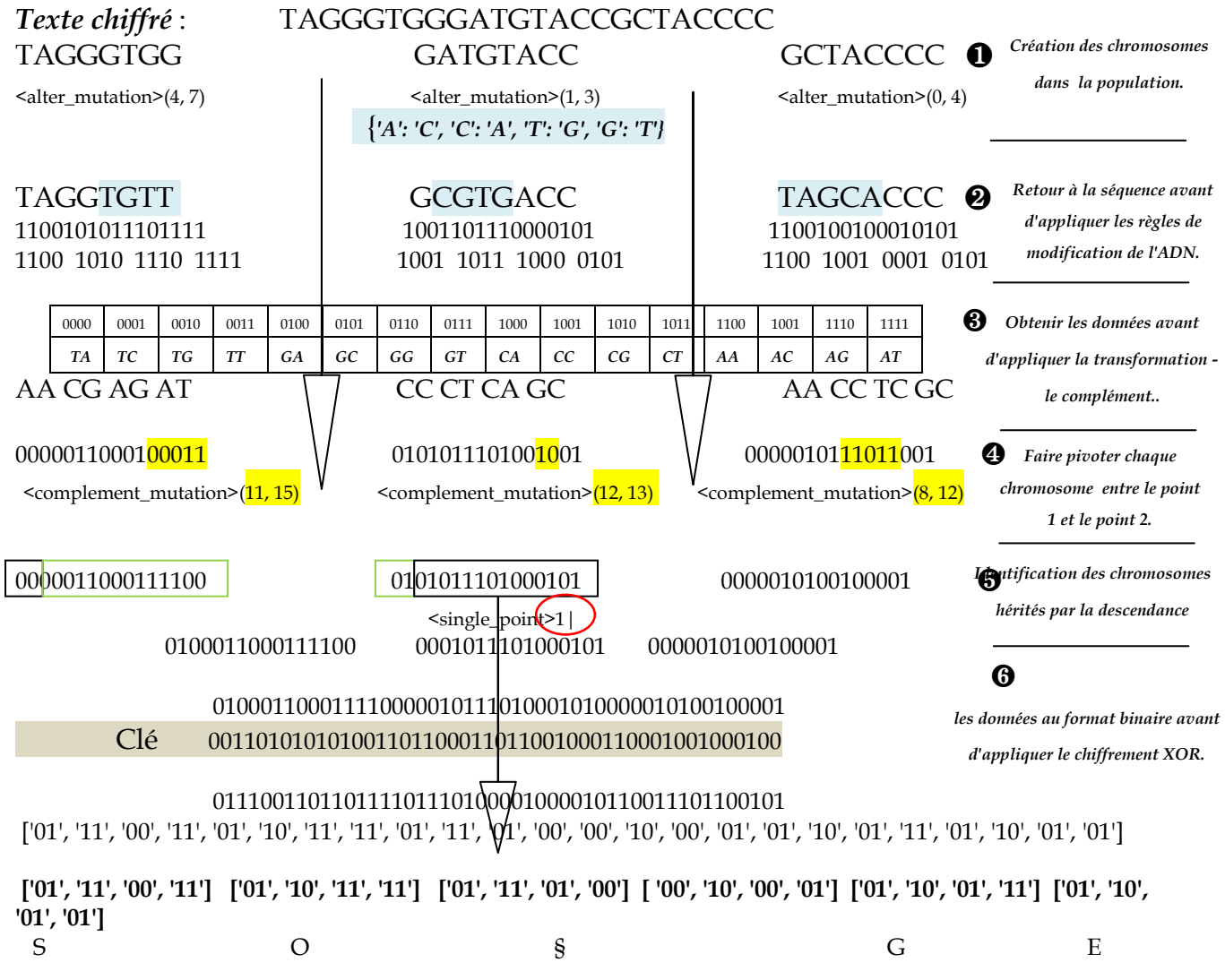
- Sur la base des données dans le tableau, on peut dire que l'algorithme utilisé est assez efficace en termes de consommation de mémoire.
- L'algorithme consomme une petite quantité de mémoire pour crypter de petits textes, tandis que la quantité de mémoire consommée augmente considérablement avec de grands textes.

3.3 Sécurité:

L'algorithme de chiffrement doit fournir un niveau élevé de sécurité pour protéger les données contre tout accès non autorisé ou modification.
Pour évaluer la sécurité, nous avons modifié un seul octet dans la clé de chiffrement et le texte obtenu était tel qu'indiqué.

Clé de déchiffrement

00110101010100110110001101100100011000100100010001001111011000010100110101110100011001110
10000110101001100110001011100000110000 1<key><no_rounds>1<no_rounds><round><reshape>8<
reshape><crossover><type>single_point_crossover<type><single_point>1|<single_point><crossov
er><mutation><mutation_table>{'A': 'C', 'C': 'A', 'T': 'G', 'G': 'T'}<mutation_table>
<chromosome><complement_mutation> (11, 15)<complement_mutation><alter_mutation>(4,
7)<alter_mutation><chromosome><chromosome> <complement_mutation>(12, 13)
<complement_mutation><alter_mutation>(1, 3)<alter_mutation> <chromosome>
<chromosome><complement_mutation>(8, 12)<complement_mutation><alter_mutation> (0,
4)<alter_mutation><chromosome>



Text original : ***codage***

En général, l'utilisation d'une clé de chiffrement et l'absence du texte original signifient que les données ont été cryptées de manière sécurisée et ne peuvent pas être déchiffrées sans la clé correcte.

4. Conclusion

L'algorithme de chiffrement a été proposé en utilisant l'acide désoxyribonucléique (ADN). Cet algorithme est basé sur le principe de la double hélice de l'ADN et sa capacité à stocker des informations en grande quantité et de manière fiable. L'algorithme a été évalué à travers une série de tests, et les résultats de l'évaluation ont montré que l'algorithme présente un niveau élevé de sécurité, une efficacité d'utilisation de la mémoire et une vitesse de chiffrement et de déchiffrement des données.

Cependant, l'algorithme de chiffrement utilisant l'ADN nécessite davantage de recherche et de développement avant d'être prêt à être utilisé. Les domaines de recherche future comprennent l'amélioration de la vitesse du processus de chiffrement et de déchiffrement, le développement de méthodes sécurisées de gestion des clés, ainsi que l'étude de l'impact de l'environnement sur les performances de l'algorithme.

Conclusion Général

Conclusion Général

En conclusion de cette étude, il est important de souligner l'efficacité du chiffrement a base de la séquence d'ADN pour assurer la sécurité des données dans l'environnement du Cloud Computing.

Les recherches ont démontré que cette technique offre un niveau élevé de sécurité et de confidentialité par rapport aux méthodes traditionnelles de chiffrement. Sa résistance aux attaques quantiques en fait une option sûre pour protéger les informations sensibles.

De plus, la facilité d'utilisation de cette technique, qui ne nécessite pas d'outils complexes, et la possibilité de stocker les données chiffrées dans des molécules d'ADN permettent d'économiser de l'espace de stockage et de garantir l'intégrité des données à long terme.

En outre, cette technologie offre des perspectives prometteuses pour le futur en permettant son intégration avec d'autres technologies telles que la blockchain pour renforcer la sécurité. De nouvelles méthodes de chiffrement et de déchiffrement des données en utilisant l'ADN peuvent également être développées pour améliorer la vitesse et l'efficacité du processus.

le chiffrement a base d'ADN est une technologie prometteuse pour assurer la sécurité des informations dans l'ère du Cloud Computing. Il est essentiel de poursuivre la recherche et le développement dans ce domaine pour exploiter pleinement le potentiel de cette technologie dans la protection des données contre les menaces actuelles et futures.

Bibliographie :

[1] Rajan, S., Jairath, A.: Cloud computing: The fifth generation of computing. Communication Systems and Network Technologies (CSNT), 2011 International Conférence on. IEEE, (2011).
[2] Sullivan, L.: Cloud Computing in Life Sciences R&D, report on InsightPharmaReports, Cambridge Healthtech Institute (CHI), 781-972-1353 (2010).
[3] Vincent Kherbache and all. Cloud Computing, IUT Nancy Charlemagne, 2009/2010
[4] http ://www.cisco.com/web/strategy/docs/gov/CiscoCloudComputing WP.pdf
[5] Tim Mather, 2009
[6] Alain-B TCHANA, système d'administration autonome adaptable : application au Cloud, L'institut national polytechnique de Toulouse , 2011
[7] Landry Fossouo Noumsi, étude et mise en place d'une solution Cloud Computing, école national supérieur des postes et des télécommunications, 2012.
[8] Mr. Youcef Lammari '' Mémoire de Fin d'études Pour l'obtention du diplôme de Master en Informatique ''(La gestion d'authentification dans un environnement Cloud)
[9] Le Cloud Computing, Définition et impact pour les SSII, 2012.
[10] https://th.bing.com/th/id/R.d8fed52ca066320f52f6891b26647226?rik=9r%2fiWeDRizQsTg&pid=ImgRaw&r=0 À 13h22 le 13 février 2024.
[11] EuroCloud France. L'évolution maitrisée vers le laaS/PaaS novembre 2011.
[12] WygwanLe Cloud Computing: Réelle révolution ou simple évolution ?
[13] le Cloud Computing une nouvelle filière fortement structurante, septembre 2012
[14] Mavault W.: Vers une architecture pair-a-pair pour l'informatique dans le nuage Thèse die Doctorat, Université de Grenoble, France, Octobre 2011
[15] "The Network Simulator Ns2 (2010). [Online] Available at :http://www.isi.edu/nsnam/ns/
[16] Philippe Hedde, Syntec numérique Livre blan sécurité de Cloud Computing Analyse des risques, réponses et bonnes pratiques, 2010
[17] Mather.T, Subra.K, Latif.S, « Cloud security and privacy », Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.,2009
[18] Li. C, "Computing complete answers to queries in the presence of limited access patterns," VLDB Journal, vol. 12, no. 3, pp. 211–227, Oct. 2003.
[19] Ardagna. C, S. De Capitani di Vimercati, S. Paraboschi, E. Pedrini, P. Samarati, et M. Verdicchio, "Expressive and deployable access control in open web service applications," IEEE TSC, vol. 4, no. 2, pp. 96–109, Apr-Jun. 2011
[20] Chen.D H.Zhao, « Data Security and Privacy Protection Issues in Cloud Computing », in Proc of International Conference on Computer Science and Electronics Engineering, 2012
[21] Sandhu. R, et P. Samarati, "Authentication, access control and intrusion detection," in CRC Handbook of Computer Science and Engineering, A. Tucker, Ed. CRC Press Inc., 1997, pp. 1929–1948
[22] Benedikt. M, P. Bourhis, et C. Ley, "Querying schemas with access restrictions," Proc. of VLDB Endowment, vol. 5, no. 7, pp. 634–645, Mar. 2012.
[23] Lee. A, M. Winslett, J. Basney, and V. Welch, "The Traust authorization service," ACM TISSEC, vol. 11, no. 1, pp. 1–3, Feb. 2008.
[24] Cimato. S, M. Gamassi, V. Piuri, R. Sassi, and F. Scotti, "Privacyaware biometrics: Design and implementation of a multimodal verification system," in Proc. of ACSAC, Anaheim, CA, USA, Dec. 2008
[26] Vimercati. V S. Foresti P. Samarati, « Managing and Accessing Data in the Cloud : Privacy Risks and Approaches », IEEE, 2012

[27] Chen. W, L. Clarke, J. Kurose, and D. Towsley, "Optimizing cost sensitive trust-negotiation protocols," in Proc. of INFOCOM, Miami, FL, USA, Mar. 2005.
[28] Ardagna. C, S. De Capitani di Vimercati, S. Foresti, S. Paraboschi, and P. Samarati, "Minimizing disclosure of private information in credentialbased interactions: A graph-based approach," in Proc. of PASSAT Minneapolis, MN, USA, Aug. 2010
[29] Yao. D, K. Frikken, M. Atallah, and R. Tamassia, "Private information: To reveal or not to reveal," ACM TISSEC, vol. 12, no. 1, pp. 1–27, Oct. 2008
[30] Karger. P, D. Olmedilla, et W.-T. Balke, "Exploiting preferences for minimal credential disclosure in policy-driven trust negotiations," in Proc. of SDM, Auckland, New Zealand, Aug. 2008.
[31] Samarati. P , S. De Capitani di Vimercati, "Data protection in outsourcing scenarios: Issues and directions," in Proc. of ASIACCS, China, Apr. 2010
[32] Aggarwal. G, M. Bawa, P. Ganesan, H. Garcia-Molina, K. Kenthapadi, R. Motwani, U. Srivastava, D. Thomas, and Y. Xu, "Two can keep a secret: A distributed architecture for secure database services," in Proc.of CIDR, Asilomar, CA, USA, Jan. 2005
[33] Ciriani. V, S. CapitaniForesti, "Fragmentation design for efficient query execution over sensitive distributed databases," in Proc. of ICDCS, Montreal, Quebec, Canada, Jun. 2009.
[34] Ciriani. V, S. Capitani S. Foresti S.Jajodia "Support for write privileges on outsourced data," in Proc. of SEC, Heraklion, Crete, Greece, Jun. 2012.
[35] Capitani. S, di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, et P. Samarati, "Over-encryption: Management of access control evolution on outsourced data," in Proc. of VLDB, Vienna, Austria, Sep. 2007
[36] M. Ilhem et G. N. E. Houda – Cryptographie homomorphe pour les réseaux «vehicular cloud computing», Mémoire, Université Abou Bekr Belkaid - Tlemcen, 2017
[37] Kaspersky – "What is data encryption? definition and explanation", Dec 2021.
[38] M. K. Adomey – "Introduction to cryptography".
[39] B. Schneider – "Cryptographie appliquée. 2 eme édition, 2001".
[40] T. Perret – "qu'est-ce que le chiffrement homomorphe ?", citation 6 (2022).
[41] D. B. . G. Dartois – "cryptographie paris 13", (2010).
[42] B. CHENENE – "Chiffrement des vidéos numériques", Thèse, UNIVERSITE MOHAMED BOUDIAF-M'SILA FACULTE DES MATHEMATIQUES ET DE L . . . , 2019.
[43] Kaspersky – "What is data encryption? definition and explanation", Dec 2021
[44] , Cryptographie homomorphe pour les réseaux «vehicular cloud computing», Mémoire, Université Abou Bekr Belkaid - Tlemcen, 2017.
[45] D. N. Kuate – "Cryptographie homomorphe et transcodage d'image/video dans le domaine chiffré", Thèse, Université Paris Saclay (COmUE), 2018.
[46] A. Vengadapurvaja, G. Nisha, R. Aarthy et N. Sasikaladevi – "An efficient homomorphic medical image encryption algorithm for cloud storage security", Procedia computer science 115 (2017), p. 643–650.
[47] A. Schools, A. B. Model, A.-B. Model, A. Colony, A. B. Colony, A. N. Network, A. Computing, B. Data, B. V. O. Local, C. Albicans et al. – "Complex adaptive systems, publication 3 cihan h. dagli, editor in chief conference organized by missouri university of science and technology 2013-baltimore, md", Procedia Computer Science 20 (2013), p. 553–558.
[48] Debnath Bhattacharyya, Samir Kumar Bandyopadhyay, ʊ"Hiding Secret Data in DNA Sequence", International Journal of Scientific &Engineering Research Volume 4, Issue 2, February-2013 ISSN 2229-5518.
[49] Abbasy, M. R., Nikfard, P., Ordi, A., & Torkaman, M. R. N. (2012). "DNA base data hiding algorithm", International Journal of New Computer Architectures and their Applications (IJNCAA), 2(1), 183-192.

[50] Cheng Guo, Chin-Chen Chang and Zhi-Hui Wang ʊ"A New Data Hiding Scheme Based On DNA Sequence" .International Journal of Innovative Computing, Information and Control ICIC International Volume 8, Number 1(A), January 2012.
[52] Mohammad Reza Abbasy, Azizah Abdul Manaf, and M.A.Shahidan, ʊ"Data Hiding Method Based on DNA Basic Characteristics". International Conference on Digital Enterprise and Information Systems, July 20-22, (2011), London, UK,pp. 53–62.
[53] Wang Zhong, Zhy Yu, "Index-based symmetric DNA encryption algorithm". Image and Signal Processing (CISP), 2011 4th International congress on image and signal processing, 15-17 oct. (2011).
[54]
https://en.wikipedia.org/wiki/Molecular_Structure_of_Nucleic_Acids:_A_Structure_for_Deoxyribose_Nucleic_Acid
[55] De notre réalisation, "les étudiants Daoudi Mohamed Amine et Semir fatima". ' Un Modèle De Chiffrement Pour Préserver La Confidentialité Dans Les Clouds ' (juin 2024)

ملخص

تنتقل المعلومات عبر الشبكة عبر مسارات مختلفة، ومن الضروري تأمينها للسيطرة على الوصول غير المصرح به إلى البيانات بجميع أشكالها. يتم ضمان أمان البيانات عن طريق تحويل النص العادي إلى شكل غير قابل للقراءة باستخدام خوارزميات التشفير. تعتمد تقنيات التشفير على تشفير الرسائل لضمان اتصال آمن عبر الشبكة. من الضروري للشركات والأفراد تأمين معلوماتهم ضد الهجمات وقراصنة الحواسيب لضمان سرية وسلامة البيانات. لتعزيز أمان البيانات على الشبكة، من الضروري تشفير البيانات بشكل غير قابل للقراءة. تم إجراء العديد من الأبحاث حول تطور أنظمة التشفير، ومع ذلك، التطور الحالي في هذا المجال هو تشفير الحمض النووي الريبوزي الذي تم تطويره بناءً على قدرة حساب الحمض النووي. تقترح البحث طريقة لتنفيذ إخفاء البيانات باستخدام سلاسل الحمض النووي. يتم استخدام مفاهيم الترميز الثنائي والقيم العشوائية، ويتم إنشاء مفتاح لتشفير البيانات. يشارك المرسل سلسلة الحمض النووي التي تمثل البيانات المشفرة المستخرجة من البيانات الأصلية. بهذه الطريقة، يتم ضمان معالجة المعلومات في الحوسبة السحابية دون تحديد البيانات الأصلية. بالإضافة إلى ذلك، قمنا بتقييم خوارزمية التشفير مع مراعاة العديد من العوامل، بما في ذلك طول المفتاح والأداء واستهلاك الموارد، لاتخاذ القرار المناسب.

***الكلمات المفتاحية**: الحوسبة السحابية،خوارزميات التشفير، سرية وسلامة البيانات، سلسلة الحمض النووي*

Abstract

Information circulates through the network via different paths, and it is necessary to secure it to control unauthorized access to data in all its forms. Data security is ensured by converting plain text into an unreadable form using encryption algorithms. Encryption techniques rely on encrypting messages to ensure secure communication across the network. It is essential for businesses and individuals to secure their information against attacks and hackers in order to guarantee data confidentiality and integrity. To enhance data security on the network, it is necessary to encrypt data in an unreadable manner. Many researches have been conducted on the evolution of encryption systems, however, the current evolution in this field is RNA encryption, which has been developed based on DNA's computing capacity. The research proposes a method for data concealment using DNA strands. Binary coding and random values concepts are used, and a key is created to encrypt the data. The sender shares the DNA strand representing the encrypted data extracted from the original data. This ensures that the processing of information in Cloud Computing is done without identifying the original data. Furthermore, we evaluated the encryption algorithm considering various factors, including key length, performance, and resource consumption, in order to make the appropriate decision.

***Keywords**: Cloud Computing, encryption algorithms, data confidentiality and integrity, DNA strand.*

Résumé

Les informations circulent à travers le réseau via des chemins différents, et il est nécessaire de les sécuriser pour contrôler l'accès non autorisé aux données sous toutes leurs formes. La sécurité des données est assurée en convertissant le texte normal en une forme illisible à l'aide d'algorithmes de cryptage. Les techniques de cryptage reposent sur le chiffrement des messages pour garantir une communication sécurisée à travers le réseau. Il est essentiel pour les entreprises et les particuliers de sécuriser leurs informations contre les attaques et les pirates informatiques afin de garantir la confidentialité et l'intégrité des données. Pour renforcer la sécurité des données sur le réseau, il est nécessaire de crypter les données de manière illisible. De nombreuses recherches ont été menées sur l'évolution des systèmes de cryptage, cependant, l'évolution actuelle dans ce domaine est le cryptage de l'acide ribonucléique qui a été développé en fonction de la capacité de calcul de l'ADN. La recherche propose une méthode pour exécuter la dissimulation des données en utilisant des chaînes d'ADN. Les concepts de codage binaire et de valeurs aléatoires sont utilisés, et une clé est créée pour crypter les données. L'expéditeur partage la chaîne d'ADN représentant les données cryptées extraites des données d'origine. De cette manière, il est assuré que le traitement des informations dans le Cloud Computing est effectué sans identifier les données d'origine. De plus, nous avons évalué l'algorithme de cryptage en tenant compte de nombreux facteurs, y compris la longueur de la clé, les performances et la consommation de ressources, afin de prendre la décision appropriée.

***Mots clés** : Cloud Computing, Les algorithmes de cryptage, la confidentialité et l'intégrité des données, la chaîne d'ADN*