Contributions à la préservation de la vie privée et de la sécurité des données partagées

Achref Aloui

N° National de Thèse : XXX

UNIVERSITÉ PARIS 8
VINCENNES-SAINT-DENIS

THÈSE

en vue de l'obtention du grade de

Docteur de l'Université Paris 8,
délivré par UNIVERSITÉ PARIS 8

Discipline : **Mathématique-Informatique**

Laboratoire Analyse, Géométrie et Applications (LAGA), UMR 7539, équipe AGC3

École Doctorale (ED 224) Cognition, Langage, Interaction

Présentée et soutenue publiquement le 28 JUIN 2022
par **Achref ALOUI**

Contributions à la préservation de la vie privée et de la sécurité des données partagées

---

Devant le jury composé de :

M.	Talel ABDESSALEM	*Professeur des Universités, Télécom Paris*	Examinateur
M.	Ridha BOUALLEGUE	*Professeur des Universités, Sup'Com Tunis*	Examinateur
Mme.	Caroline FONTAINE	*Directrice de Recherche CNRS, ENS Paris-Saclay*	Rapporteuse
M.	Patrick LACHARME	*Maître de Conférences HDR, Ensicaen*	Rapporteur
Mme.	Sihem MESNAGER	*Maître de Conférences HDR, Université Paris VIII*	Directrice
M.	Abdellah MOKRANE	*Professeur des Universités, Université Paris VIII*	Examinateur

Laboratoire d'Analyse, Géométrie et
Applications (LAGA),UMR 7539,
équipe AGC3
Adresse
99 Av. Jean Baptiste Clément,
93430 Villetaneuse, France

École doctorale (ED 224) Cognition,
Langage, Interaction
2 Rue de la Liberté,
93200 Saint-Denis, France

Remerciements

Tout d'abord, j'adresse ma sincère gratitude à ma directrice de thèse Mme Sihem MESNAGER, que je connais depuis mes débuts en cryptographie lors de son cours "initiation à la cryptographie" en Licence 3 et qui m'a beaucoup aidé non seulement sur tous les aspects scientifiques en m'apprenant la rigueur , le travail méthodique et le développement des idées pour trouver des résultats mais aussi sur le plan humain. Je ne la remercierai jamais assez d'avoir toujours été là à mes côtés pour m'aider avec autant de patience dès mes premiers pas dans la recherche sur des sujets orientés mathématiques avant de passer à des aspects plus applicatifs dans lesquels j'éprouve plus de plaisir. Sa générosité, son approche attentive et sa manière délicate et gracieuse d'aborder toutes les difficultés scientifiques et de vie que j'ai rencontrées ont eu une influence importante dans ma vie professionnelle et personnelle, dont je profiterai toute ma vie.

J'exprime toute ma gratitude à Mme Caroline FONTAINE et à Mr. Patrick LACHARME pour me faire l'honneur d'être rapporteurs de cette thèse et pour le temps minutieux accordé à leur lecture approfondie ainsi que leurs recommandations précieuses et appropriées.

Je suis également très reconnaissant envers Monsieur Talel ABDESSALEM et Mr. Stéphane BRESSAN, de me m'avoir fourni un cadre de travail idéal et de m'avoir aidé à Telecom Paris dans ce travail de recherche, qui été soutenu par le projet IDOLE ANR en France, la Fondation nationale de la recherche à Singapour, le Cabinet du Premier ministre de Singapour dans le cadre de son Corporate Laboratory@University Scheme, l'Université nationale de Singapour et Singapore Telecommunications Ltd. Je suis extrêmement heureux et honoré que Mr. Talel ABDESSALEM ait accepté de faire partie de mon jury de thèse.

Je remercie vivement Mr. Ridha BOUALLEGUE et Mr. Farid MOKRANE pour m'avoir honore en acceptant de faire partie de mon jury de these. Je garde un très bon souvenir de Mr. MOKRANE de l'epoque quand j'étais etudiant en Master.

Je remercie chaleureusement Mme Mounira MSAHLI, Maitre de conférences à Télécom Paris, pour sa bienveillance, sa disponibilité et ses conseils appropriés.

Je n'oublierai jamais Mr. Gérard D. Cohen, ancien professeur à Télécom Paris, qui m'a

fait preuve de tant d'affection et tant d'encouragements. Je suis attristé par sa disparition en Juin 2018. Il restera toujours dans nos coeurs et nos pensées.

Enfin et surtout, je remercie toute ma famille et plus particulièrement ma femme Sameh pour son soutien sans faille avec si grande affection. Elle restera pour toujours ma plus grande richesse et la plus merveilleuse source qui me donne tant de force. Merci milles fois d'avoir toujours ete de mes côtés. Je profite de cette belle occasion pour exprimer à quel point j'ai été heureux d'avoir ma fille Ellen dans ma vie durant ce parcours doctoral, sa présence a aussi contribue à atténuer des difficultés de la vie durant toutes ces années.

Abstract :
Nowadays, many challenges arise in privacy due to the rapid increase in the volume of sensitive data, the need to extract it from the analyzer, and to identify it when sharing in distributed systems. Typically, the Big Data field was born to take up this kind of challenge in a context where the orders of magnitude are immense.

In this Ph.D. thesis, we are particularly interested in anonymization and security when sharing sensitive and private data. Firstly, we present a new protocol (PPDS) to preserve confidentiality in a distributed system. We are mainly focused on providing solutions to the following specific issues at the node level (e.g., a bank, but it could be other structures like a hospital) that process sensitive data :

(a) How to aggregate the records recorded in the various branches of the Bank while protecting the confidentiality of clients without the intervention of a trusted third party in the process ;

(b) How to merge data stored in separate bank branches while maintaining customer privacy.

Secondly, we improve the performance of a model regarding anonymizing sensitive data making it very difficult to identify their private users. Adequate data anonymization is indeed essential for big data analysis while preserving user privacy. Thus a company will have the capacity to exchange and communicate the data it collects through its divisions and its network of companies and partners. All data collected, as well as cross-references created in its aggregation, remain confidential. We apply our results in the specific context of NetFlow. The approach we offer consists of an analysis method that proposes classifying identifiers according to their degree of criticality. Concretely, we introduce a risk analysis phase concerning critical user identifiers to boost the anonymization model (K-anonymity) defined by Sweeny in 2002.

Résumé :
De nos jours, de nombreux défis se posent dans le cadre de la vie privée (Privacy, en anglais) en raison de l'augmentation rapide du volume de données sensibles, de la nécessité de les extraire de l'analyseur, et de les identifier lors du partage dans des systèmes distribués. Typiquement, le domaine des données massives (« Big Data ») est né pour relever ce genre de défi dans un contexte où les ordres de grandeur sont immenses.

Dans cette thèse, nous nous intéressons particulièrement à l'anonymisation et à la sécurité lors du partage de données sensibles et privées. Tout d'abord, nous présentons un nouveau protocole (PPDS) pour préserver la confidentialité dans un système distribué. Nous nous concentrons plus particulièrement sur la fourniture de solutions aux problèmes spécifiques suivants au niveau des nœuds (par exemple, une banque mais il pourrait s'agir d'autres structures comme un hôpital) qui traitent des données sensibles :

(a) Comment agréger les enregistrements enregistrés dans les différentes succursales de la Banque tout en protégeant la confidentialité des clients sans l'intervention d'un tiers de confiance dans le processus ;

(b) Comment fusionner les données stockées dans des succursales bancaires distinctes tout en préservant la confidentialité des clients.

Deuxièmement, nous améliorons les performances d'un modèle concernant l'anonymisation des données sensibles rendant très difficile l'identification de leurs utilisateurs privés. Une anonymisation adéquate des données est en effet essentielle pour l'analyse des mégadonnées tout en préservant la confidentialité des utilisateurs. Ainsi une entreprise aura la capacité d'échanger et de communiquer les données qu'elle collecte à travers ses divisions et son réseau d'entreprises et de partenaires. Toutes les données collectées, ainsi que les références croisées créées dans son agrégation, restent confidentielles. Nous appliquons nos résultats dans le contexte spécifique de NetFlow. L'approche que nous offrons consiste en une méthode d'analyse qui propose une classification des identifiants selon leur degré de criticité. Concrètement, nous introduisons une phase d'analyse des risques concernant les identifiants utilisateurs critiques pour dynamiser le modèle d'anonymisation (K-anonymity) défini par Sweeny en 2002. .

Publications

Les travaux de recherche présentés dans cette thèse ont été publiés dans des conférences internationale. Ci-dessous une liste de publications sélectionnées :

— Aloui, Ashref, et al. "Privacy as a Service : Anonymisation of NetFlow Traces." International Conference on e-Business Engineering. Springer, Cham, 2019.

— Aloui, Ashref, et al. "Preserving privacy in distributed system (PPDS) protocol : Security analysis." *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC). IEEE, 2017.*

— Aloui, Ashref, et al. "Preserving privacy in distributed system (PPDS) protocol." *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC). IEEE, 2017.*

Table des matières

Introduction

0.1 Contexte et Objectif de la thèse

Les avancées technologiques des deux dernières décennies ont abouti à une disponibilité sans précédent des données. Ceci est en grande partie la conséquence de l'amélioration continue du matériel informatique, de l'évolution rapide des technologies de gestion des données et de l'explosion de la collecte des données qui en résulte. Les progrès en matière de matériel ont fourni un stockage persistant massif, des réseaux extrêmement rapides et des processeurs puissants. Ces capacités techniques ont permis une explosion de la collecte de données. Il existe maintenant des enregistrements électroniques de nombreux événements dans des plate-formes de stockage : le cloud et Big Data, etc. On trouve par exemple, les données sur les individus collectées à travers des traces de transactions financières, des appels téléphoniques et des historiques de navigation sur le Web.

La disponibilité sans précédent des données offre sans aucun doute une grande promesse. En effet, l'information dont nous avons besoin est souvent facile à trouver. Nous pouvons combiner et agréger des données pour obtenir de nouvelles informations. Nous pouvons aussi partager et échanger des informations pour permettre le traitement collaboratif entre plusieurs parties.

Un danger peut découler de l'utilisation abusive des données et peut causer des dommages. Malgré les récentes avancées dans la gestion des données, le risque de perte ou de divulgation reste valide. Par exemple, près de 50 millions de documents à des sociétés, des organismes gouvernementaux ou des établissements universitaires contenant des données sensibles sur des personnes ont été perdus ou volés. [1]. Un travail de recherche sur le respect de la vie privée des utilisateurs a démontré un exemple plus subtil des dangers possibles. Le travail consiste à agréger deux bases de données publiques, chacune est jugée sûre pour être divulguée isolément. Cela a permis de révéler le diagnostic médical d'un politicien. [2].

L'échange et le partage d'un volume énorme des données sensibles de différentes sources, avec l'approche Big Data et "Database-as-a-Service" (DaS) [3] posent une question fondamentale : comment préserver la vie privée des utilisateurs des données externalisées lors du traitement ou d'échange ?

Dans cette thèse, nous adressons le problème d'échange des données par des sources géographiquement dispersées. Selon le livre "*Distributed Systems Security : Issues, Processes and Solutions*" , la sécurité des systèmes distribués désigne un vaste ensemble de

politiques et de technologies déployées pour protéger les données et les infrastructures associées contre différentes menaces telles que la perte de données, l'accès non autorisé aux données, la violation des données, le déni de service, etc. En général, ces menaces peuvent être regroupées en trois principaux modèles, selon le type d'attaquant qui a pour but d'accéder aux données ou de les altérer :

— **Le modèle honnête-mais-curieux** (honest-but-curious), aussi appelé semi-honnête où l'attaquant suit les étapes prescrites dans un protocole. Cependant, l'attaquant peut essayer de trouver des informations supplémentaires sur les utilisateurs, ainsi que sur les entrées, les sorties et les messages reçus pendant l'exécution du protocole sécurisé.

— **Le modèle malveillant** (malicious) où un attaquant peut changer arbitrairement l'exécution classique d'un protocole et modifier les données de l'utilisateur.

— **Le modèle déguisé** (covert) qui se situe entre le modèle honnête-mais-curieux et le modèle malveillant. Plus précisément, un attaquant déguisé peut changer arbitrairement des règles d'un protocole, sans modifier les données de l'utilisateur.

Dans cette thèse, nous supposons que notre modèle est honnête mais curieux. Ce modèle est bien adapté au problème de traitement des données dans un système distribué. Toutefois, cela peut porter atteinte à la vie privée des utilisateurs. La préservation de la vie privée ou la confidentialité des données sont les principales exigences dans un environnement distribué. Les utilisateurs des systèmes distribués s'attendent à cacher leurs identités en utilisant les mécanismes appropriés. Avec le nouveau règlement général sur la protection des données (RGPD), la préservation de la vie privée des utilisateurs des données critiques et sensibles est devenue indispensable. Les données doivent être masquées ou chiffrées et seuls les utilisateurs autorisés ont accès.

Dans cette thèse, nous concentrons notre travail de recherche sur l'échange et le croisement d'un volume énorme des données classées privées situées sur plusieurs sites dispersés géographiquement.

Un autre aspect qu'on ne peut pas négliger est l'étude de performance des systèmes distribués. En fait, un système distribué est constitué de plusieurs sites (ou centres de données), chacun situé à un endroit géographique différent. Un centre de données (data center) dispose de ses propres ressources informatiques et de stockage, généralement sous la forme d'un cluster d'ordinateurs, qui est composé d'un grand nombre de nœuds de calcul/stockage. Ainsi, pour des raisons d'évolutivité et de performance, les données peuvent être réparties sur plusieurs nœuds, soit horizontalement (différents éléments de données sur différents nœuds), soit verticalement (différents champs de données sur différents nœuds). Ces deux formes de partitionnement peuvent être combinées. Dans cette thèse, nous considérons deux cas pour le traitement de la vie privée des utilisateurs des données :

1. **Distribué** : Les données sont réparties sur plusieurs nœuds.
2. **Centralisé** : Les données des utilisateurs sont stockées sur un seul nœud dans un centre de données.

0.2 Contributions

Ce manuscrit présente deux contributions :

Contribution 1 : Nous proposons un protocole complet, appelé PPDS (Protocol for Preserving Privacy in Distributed System). L'objectif du protocole PPDS est d'assurer la protection de la vie privée dans un environnement distribué pour un volume énorme des données 'Big Data'. Le scénario considère plusieurs agences bancaires qui souhaitent partager le traitement des données tout en protégeant la confidentialité des données. Une preuve de concept avec une analyse de sécurité est également traitée.

Contribution 2 : Nous fournissons une amélioration d'un paradigme original d'anonymisation des données afin de rendre impossible la ré-identification des utilisateurs associés. Nous considérons le cas d'un journal NetFlow comme source des données de test. La solution comprend un processus d'analyse des risques pour classer les identifiants en fonction de leurs niveaux de criticité. Nous utilisons un paradigme dynamique d'anonymisation avec évaluation des performances.

Ces contributions sont publiées dans des conférences classe B, listées ci-dessous :

— Aloui, Ashref, et al. "Protocol for preserving privacy in distributed system (PPDS)." *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC). IEEE, 2017.*

— Aloui, Ashref, et al. "Preserving privacy in distributed system (PPDS) protocol : Security analysis." *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC). IEEE, 2017.*

— Aloui, Ashref, et al. "Privacy as a Service : Anonymisation of NetFlow Traces." International Conference on e-Business Engineering. Springer, Cham, 2019.

0.3 Le plan de la thèse

Cette thèse est organisée comme suit.

Partie 1 : État de l'art

Chapitre 1 : La sécurité des systèmes distribués et leurs données

L'objectif de ce chapitre est d'exposer le fonctionnement des systèmes distribués et la façon dont leurs mécanismes présentent des défis de sécurité et de la préservation de la vie privée. On s'attend très souvent à ce que les techniques de sécurité classiques s'appliquent directement dans ces systèmes toutefois, ce n'est pas le cas. En effet, on a besoin de mieux comprendre la base conceptuelle d'un système distribué pour bien assurer sa sécurité. Ce chapitre aborde la catégorisation fonctionnelle des systèmes distribués en deux grandes classes : le contrôle décentralisé et coordonné.

Chapitre 2 : Systèmes distribués : modèles, sécurité et respect de la vie privée

Ce chapitre traite des préoccupations relatives à la vie privée et à la sécurité des données massives. Il traite l'utilisation de la préservation de la vie privée en prenant compte des méthodes existantes telles que k-anonymat, T-proximité et L-diversité et la Confidentialité différentielle . Un certain nombre de mécanismes de préservation de la vie privée ont été mis au point (Par exemple : la génération de données, le stockage des données et le traitement des données) d'un cycle de vie des données volumineuses. Ce chapitre présente également un état de l'art sur les techniques récentes de préservation de la vie privée pour les données massives.

Partie 2 : Contributions

Chapitre 3 : Contribution 1

Dans ce chapitre, nous présentons une solution qui assure l'anonymisation des données sensibles échangées entre des sites honnêtes mais curieux lors du traitement de manière anonyme sans avoir besoin à un tiers de confiance impliqué dans le processus.

Dans ce chapitre, nous présentons également la mise en œuvre de la préservation de la vie privée dans le protocole de système distribué (PPDS). Nous avons analysé sa sécurité et nous avons proposé une solution de détection des attaques DoS en utilisant le concept d'entropie. Le temps de réponse et le temps de traitement des données sont également analysés en tant que résultats de performance de notre preuve de concept.

Chapitre 4 : Contribution 2

Dans ce chapitre, nous présentons une solution d'anonymisation dynamique en utilisant en amont une évaluation de la criticité des identifiants des utilisateurs. Le schéma classe les données en fonction de leur niveau de criticité. En fonction de ce niveau, nous fixons un facteur d'anonymisation K pour chaque groupe de données. La fonction de généralisation est également choisie en fonction du niveau de criticité des données. Le prototype implémenté a été testé avec des traces NetFlow. Les résultats de simulation de notre solution montrent que la répartition des données change complètement en appliquant un schéma de K-anonymat optimal et dynamique.

Chapitre 5 : Conclusion

Dans ce chapitre, nous résumons nos contributions.

Chapitre 1

La sécurité des systèmes distribués et leurs données

1.1 Introduction

Un système distribué est un ensemble d'entités disparates et indépendantes qui fonctionnent pour un objectif commun. Il garantit un accès fiable et immédiat avec une haute disponibilité pour assurer des services. Ce système est basé sur la redondance et la coordination des ressources afin de fournir une vue logique centralisée.

Les systèmes distribués se reposent sur les facteurs suivants afin d'atteindre ses capacités [4–8] : le flux des données échangées via les entrées-sorties, la transmission des ressources entres les éléments distribués, la coordination des ressources et les services via un Workflow et finalement la gestion des ressources via la virtualisation et la gestion du temps et les événements en exécution. Ces facteurs ont permis au système distribué d'assurer le fonctionnement des applications, des base données. Par conséquent des mesures de sécurité, on été mises on place afin de faire face aux différentes menaces probables au système. Un mécanisme de contrôle d'accès et d'admission pour les éléments composants du système, des protocoles et des applications middleware distribuées assure la disponibilité des ressources via la réplication sur plusieurs sites.

Ce chapitre présente tout d'abord les différentes classes des systèmes distribués ensuite, il aborde progressivement les différentes menaces pour le système et les services distribués décentralisés.

1.2 Les classes de systèmes distribués

D'après [4, 9] les systèmes distribués peuvent être classés en plusieurs catégories suivant leurs caractéristiques et leurs déploiements. Par exemple, le cloud et les systèmes Peer to Peer sont des exemples de systèmes distribués définis suivant l'agrégation de leurs ressources réparties, ou via les services de base des systèmes distribués comme les bases de données (MariaDB Galera), ou à travers les logiciels de middleware comme WildFly.

Généralement, les systèmes distribués peuvent être classés sous deux grandes catégories.

1. Les éléments du systeme sont coordonnés via un service ou des ressources distribuées, le cloud public privé et hybride (Vms), les services de gestions des données Google File System, AWS et AZURE et les services pour le Big Data et Wildfly, sont des exemples de systèmes distribués avec des ressources géo-dispersées.
2. Les éléments du système sont indépendants comme les services de partage de fichiers (BearShare, LimeWire, KaZaa, eMule, Vuze, uTorrent and BitTorrent) qui sont nommés des systèmes Peer to Peer, un service sans gestion centralisé.

Dans la littérature, Hamid et al [10] mettent l'accent sur les concepts et les mécanismes de sécurité dans un système distribué où les ressources et les services sont dispersés.Tbatou et al, [11] considèrent l'utilisation de la distribution comme un mécanisme de sécurité pour éviter les points de défaillance et assurer la continuité de service en cas de problème. On peut citer comme exemple, la gestion distribuée des clés pour des machines virtuelles (VM) afin d'isoler les ressources et les applications OS.

La partie suivante met l'accent sur la sécurité dans un système distribué. Elle examine également les mécanismes de sécurité dispersés qui s'exécute généralement avec les ressources dispersées, ce qui conduit directement à la nécessité de la coordination des ressources et les menaces. Une architecture d'un système distribué est souvent une agrégation de plus qu'une couche qui combine les pièces informatiques telles que la mémoire, les systèmes de gestion de fichiers, les protocoles de communication et les connexions de middleware.

1.2.1 La sécurité dans les systèmes distribués

Toute faiblesse dans la conception et l'implémentation d'un système informatique produit des failles de sécurité qui mènent dans la plupart du temps à des attaques informatique. Dans un système distribué un attaquant peut exploiter les vulnérabilités des éléments fonctionnels du système telle que la partie physique, les différents protocoles utilisés pour échanger et synchroniser les données et l'ensemble des services.

Nous décrivons dans ce qui suites différentes mesures de sécurité pour chaque facteur fonctionnel du système.

Le contrôle d'accès

Assurer la sécurité du flux de données d'entrée/sortie via la gestion de contrôle d'accès. Ce mécanisme est la partie qui assure à tout intervenant externe autorisé à accéder à des ressources ou à la lecture et à l'écriture des données. Ce mécanisme de sécurité est vulnérable contre les attaques de déni de service qui réduit, affecte la disponibilité et l'intégrité du système. L'attaquant peut intercepter, falsifier ou voler une identité d'une entité autorisée, cette entité peut être morale ou physique ses droits d'accès peuvent accéder à des ressources ou à des éléments du système. Une identité peut être un simple mot de passe ou un schéma d'autorisation basé sur des privilèges suivant l'autorisation et le type de connexion.

La transmission de données

La transmission des données dans les supports réseau entre les éléments du système tel que les messages échangés entres différents composants du système, le routage et les événements interne du système peuvent être menacé par deux méthodes, soit active avec la modification des donnée échangées ou passive par une simple écoute et collecte des données. [4, 12, 13]

Services de gestion et de coordination des ressources

La coordination entre les ressources représente la partie la plus critique du système. Le système coordonne ses opérations internes via des services et des protocoles. Un attaquant peut menacer le protocole de gestion des réplications et la pile des événement et la priorité lors de fonctionnement.

Sécurité des données

Les données générées, traitées ou stockées dans un système distribué sont aussi menacées par les vulnérabilités classiques. Un adversaire peut viser la disponibilité l'intégrité et la confidentialité des données, par exemple l'attaque par canaux auxiliaires [14] consiste à extraire les données via l'interception des signaux émis par le système tel que la consommation d'électricité même le bruit et le temps d'exécution, cette attaque affecte la confidentialité. Les attaques de déni de service cause un arrêt ou un retard dans le traitement des données ce qui implique une menace sur la disponibilité. Le vol des accès aux données entame une menace à l'intégrité des données.

1.3 La coordination des resources :

La coordination des ressources peut être gérée par un gestionnaire ou un orchestrateur des ressources ou des services distribués. Il donne une vue logique d'un système centralisé et contrairement au système distribué décentralisé comme le système Peer-to-Peer.

Le cloud permet au client d'accéder à une infrastructure composée par des serveurs et des ressources distribués et géographiquement dispersé, orchestrer par un coordinateur de services.

Le système Peer-to-Peer offre des services distribués, les différents éléments composants du système interagissent indépendamment afin de fournir un service purement distribué.

D'après [4] il y a deux grandes classes des systèmes distribués, les classes de coordination des ressources et la classe de coordination des services, sur la base de leur schéma de coordination bien que leurs fonctionnalités

Dans la partie suivante, tous d'abord, nous détaillons les différents styles d'orchestration dans le concept distribués des systèmes.

1.3.1 Les différents types de coordinations

La synchronisation offre au système distribué une interaction cohérente entre ses différents éléments. La synchronisation peut être suivant le temps ou suivant un ordre logique spécifique des processus et sur tous les niveaux, réseau, protocole et service.

Il existe trois types de synchronisation :

Pour que les ressources et les services distribués interagissent de manière significative, la base de synchronisation entre eux, en temps physique ou dans l'ordre logique, doit être spécifiée. La synchronisation s'applique à la fois au niveau du réseau et du processus.

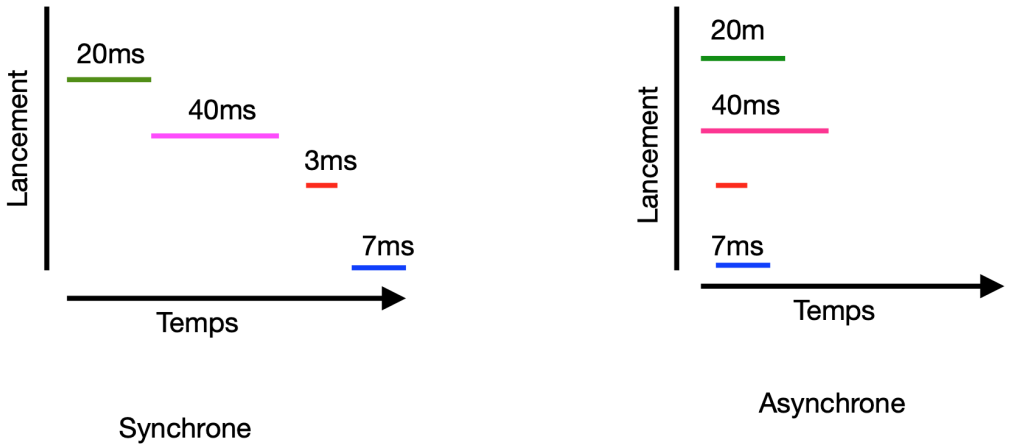


FIGURE 1.1 – la synchronisation synchrone et asynchrone

1. **Synchrone** C'est la coordination en fonction du temps. Quand une tache est en exécution la suivante doit attendre la fin de la précédente pour qu'elle s'exécute.
2. **Asynchrone** : Les composants ne se coordonnent pas en fonction du temps. Chaque tache s'exécute suivant des mesures différentes aux autres taches tant qu'elle n'influence pas sur la cohérence du système..
3. **Partiellement synchrone** : Les taches s'exécutent en mode asynchrone, s'il n'y a pas de restrictions critiques peuvant être appliqués lors d'actions.

1.3.2 Assurer une communication fiable et sûre

La communication dans un système distribué est nommée la communication du groupe [4]. La communication du groupe peut être un simple message direct d'un élément à un autre avec un accusé de réception (ACKS et NACKS), ou une multidiffusion. Pour avoir une communication du groupe et des canaux sécurisés, des cryptosystemes et des algorithmes de distribution des clé et des certificats électroniques peuvent être utilisés, mais cela entraîne une augmentation du coût de transmission.

1.3.3 Propriétés de coordination

Afin de donner une vue logique centralisée pour un système distribué avec des ressources geo-dispersées. Le système doit assurer les notions de base suivantes : *le consensus*, *la cohérence* .

Consensus

Consensus est l'obtention d'un accord sur des valeurs. Les valeurs peuvent être des données, des identificateurs des processus. Le consensus exige que les propriétés suivantes soient présentes :

1. *Accord* : Toutes les taches sont d'accord sur une même valeur.
2. *Validité* : La valeur accordée est valide.
3. *Terminaison* : Au final, une conclusion est faite.

Cohérence et composition du Groupe

L'adhésion représente un terme clé dans les systèmes distribués coordonnés. Elle représente l'accord obtenu suite à un ensemble de validation afin d'être un élément statique, dynamique ou un membre du quorum du système. La cohérence du système représente l'intégrité du service fourni. Les différents type de cohérence sont :

— ***les modelés avec une forte cohérence*** : Du point de vue sécurité, cela concerne souvent l'intégrité du service. La cohérence présente diverses nuances et les principaux types sont énumérés ci-dessous. Les éléments du système doivent observer le même ordre d'actions.
  — *Cohérence strict* : il y a aucune contrainte sur l'ordre observé des actions tant qu'il est cohérent pour tous les participants.
  — *Linéarisation* : est essentiellement une cohérence stricte avec la contrainte supplémentaire que l'ordre des actions observé correspond à leur ordre en temps réel.

  Dans le cas où les données traitées pourraient être à risque et la perte peut entraîner des conséquences désastreuses les modèles à la forte cohérence sont utilisés. La cohérence représente une importance que la disponibilité. Par exemple dans les systèmes de base données relationnelles comme MariaDB Galera et les bases de données moderne comme MongDB sont des systèmes à forte cohérence.

— ***les modèles avec une faible cohérence*** : Ses éléments peuvent ne pas suivre l'ordre des événement du service ce qui produit les états d'incohérence suivant les contraires supplémentaires que les ordres suivis doivent être satisfait. Ce qui entraîne à des états d'incohérence qui seront pris par les mécanismes de résolution de conflits [4, 15].
  — *La cohérence séquentielle* : Les taches qui sont produites par un processus du service s'exécute dans leur ordre d'origine de façon séquentielle.
  — *La cohérence causale* : Les taches qui sont dépendantes ou causalement liées doivent être préservées. Deux taches causalement liées aux mêmes effets s'ils accèdent les deux en même temps aux ressources demandée que l'un des deux aura lancé.

— *La cohérence éventuelle* : Les tâches exécutées dans l'ordre sans avoir des contraintes spéciales à satisfaire. Les différents éléments du système doivent exécuter leurs taches dans l'ordre et dans le cas d'un état non-cohérent un mécanisme de résolution de conflits doit être exécuté.

Le système Cassandra de Facebook est un exemple de systèmes avec une faible cohérence. Cassandra accueille un très grand nombre d'internautes, ce qui produit un sacrifice de cohérence afin d'atteindre la haute disponibilité du contenu pour chaque utilisateur.

1.4 Les classes de coordination et menaces

1.4.1 Perturbations

Un système distribué est menacé par différents types d'attaque, des attaques qui nuisent directement à la disponibilité des ressources par un déni de service de différentes ressources. La saturation des canaux de transmission reliant les différents composantes du système ce qui produit un impact sur la confidentialité et l'intégrité du système en global.

La mauvaise manipulation et la mauvaise conception d'un système distribué représentent les points d'entrées des attaquants et ils représentent les vulnérabilités à exploiter.

Les perturbations dans un système distribué peuvent être une conséquence d'une mauvaise manipulation ou d'une mauvaise conception qui constitue des cibles à exploiter par les attaquants. Nous classifions les différents types de perturbation en perturbation de transmission des données la perturbation au niveau de communication qui peut être regroupée de la façon suivante :

Premièrement *à base timing* :Il peut être l'envoi précoce ou retardé ou de façon désordonnée des données. Les incidents d'envoi et de réception et l'attaque DOS sont des facteurs pour perturber la communication ce qui produit un bloquage des supports de transmissions.

Deuxièmement *à base information* : Par exemple les attaques de sniffing de spoofing, phising les attaque par canaux auxiliaires et la modification des contenus, une entité malveillante dans le système peut émettre des mélanges d'information correct et incorrect a différents éléments en abusant les services de contrôle d'accès de coordination et de transmission afin de détourner le système.

1.4.2 Les attaques et implications

Dans cette partie, nous prenons les attaques possibles pour les systèmes distribués avec coordination de ressources et les systèmes distribués avec coordination de services.

La sécurité d'un système informatique s'articule autour des trois terminologies classiques suivantes : la confidentialité, l'intégrité et la disponibilité.

Comme cité dans la section précédente les attaquant qui vise le système distribué menacent tout d'abord l'infrastructure par ses ressources et le contrôle d'accès et les protocoles de transmission de données.

La securité dans la coordination des ressources

Nous prenons quelques scénarios d'attaques au système distribuées avec coordination des ressources et nous décrivons les approches d'atténuation.

Attaquer les ressources : c'est compromettre les ressources essentielles du système.

Pour éviter cette attaque il est nécessaire d'implémenter un mécanisme de contrôle d'accès externe aux différentes ressources et services du système et limiter les accès avec les droits d'accès (ACL). Utiliser une approche qui effectue la gestion et la protection des ressources comme le sandboxing. Les données échangées entre les éléments du système peuvent être également vue comme des ressources. Par conséquence, des techniques cryptographiques peuvent être utilisées.

La violation d'accès : un large éventail de catégories d'attaque telle que le masquage, le vol des identités et le compromis de gestion des identités. L'intégrité et la confidentialité des données et des services seront affectées et l'impact sur la ressource affectent la disponibilité.

Pour se protéger, des systèmes de détection d'intrusion doivent être utilisés. Un contrôle aléatoire avec des requêtes d'authentification. La vérification périodique de l'état du système pour valider les IDs.

L'attaque sur les machines virtuelles : c'est d'utiliser les attaques par canaux auxiliaires et les attaques de sniffing afin de récolter un maximum d'informations fuitées. Ces attaques compromettent l'intégrité et la confidentialité des services assuré par les machines virtuelles

Pour protéger les machines virtuelles contre les fuites des données possibles. Il faut tout d'abord détecter et localiser à quel niveau les fuites se produisent dans le système. Sachant que les attaques par canaux cachées interceptent les signaux produits au niveau matériel. Au niveau système des machines virtuelles, la gestion de violations est traitée par les techniques de stress formelles et expérimentales afin de renforcer tout le niveau.

Le compromis de la surveillance : le triade disponibilité, intégrité et confidentialité sera compromis si des informations incorrectes fournis sur l'état du système et ses services.

Pour assurer la cohérence des états des mécanismes typiques sont utilisés. La réplication et la coordination sont deux concepts de base dans les approches de protection contre le compromis de la surveillance et pour éviter les interruptions du service.

La securité dans la coordination des services

Dans cette partie, nous prenons quelques scénarios d'attaques au système distribuées avec coordination des services et nous décrivons les approches d'atténuation.

La compromission de la distribution des clés : est le compromis du processus responsable de l'authentification et la distribution des clés ce qui affecte l'intégrité et la confidentialité du service fournie.

Compromis des données au Rest : cela est analogue à la rupture des ressources dans le modèle de coordination des ressources applicable aux systèmes de stockage.

Compromis des données en transaction : à des conséquences de latence qui affecte l'intégrité des services.

— Les transactions rapides (Stockage, KVS) : sont basé sur la cohérence et la faible latence, si une latence est affectée le service perd sa disponibilité, même l'attaque de déni de service ne peut pas compromettre la cohérence du système.

— Les transaction lourde comme le blockchain et les ledgers se repose sur l'intégrité des transaction qui est la propriété principale a assurer.

Pour que le système soit solide, deux aspects doivent exister : chiffrer les communications et un consensus résistant aux attaques [4, 9] sur lequel les éléments du système doivent être d'accord sur l'historique commun des transactions. Le compromis de ces deux derniers implique la compromission du système cryptographique.

Pour les compromettre, vous devez compromettre les clés de chiffrement et les hachages stockés. Bien que ces systèmes soient théoriquement sûrs, ils peuvent s'avérer vulnérables aux nouvelles technologies telles que l'informatique quantique. En raison des exigences de la preuve de travail, les attaques de complot peuvent être très coûteuses sur un système de taille suffisante, mais peuvent être réalisables sur un système avec un petit nombre de participants.

1.5 Conclusion

Ce chapitre montre le fonctionnement des systèmes distribués ainsi que leurs structures qui peuvent avoir des contraintes de sécurité différentes. Parmi les exemples des systèmes distribués, on trouve le P2P et le Big data qui feront l'objet du chapitre suivant.

Chapitre 2

Systèmes distribués : modèles, sécurité et respect de la vie privée

2.1 Introduction

Les systèmes distribués fournissent des solutions efficaces pour le partage de données qui peuvent évoluer jusqu'à de très grandes quantités de données avec un très grand nombre d'utilisateurs. Le traitement d'un volume énorme des données sensibles en ligne par des communautés professionnelles (par exemple, banques, médical ou de recherche) devient populaire en raison des besoins croissants en partage de données. Les systèmes distribués offrent des caractéristiques intéressantes comme l'évolutivité, la distribution et l'autonomie mais des garanties limitées concernant la préservation de la vie privée (privacy) des données. Ces systèmes peuvent devenir hostiles vu la sensibilité des données partagées par des pairs semi honnêtes.

Les mécanismes de sécurité traditionnels ne parviennent pas à gérer et assurer la confidentialité des données en raison de leur volume important, de leur variété. La protection de la vie privée est l'une des questions les plus importantes [16], en effet ces données massives comprennent généralement des informations sensibles et privées. Par conséquent, l'analyse de 'Big Data' non sécurisée peut entraîner l'exposition d'informations personnelles identifiables (PII) à des risques de divulgation et de perte.

Des méthodes traditionnelles comme la cryptographie peut être utilisées, mais elles ne se révèlent pas efficaces en raison du volume énorme des données [17]. Pour anonymiser ces données on utilise des processus de modification empêchant l'identification des utilisateurs [18].

Ce chapitre présente les différentes manières d'organiser un réseau de nœuds purement distribué, ensuite nous explorons les différentes attaques auxquelles ces systèmes sont confrontés. Ce chapitre traite aussi les mécanismes de préservation de la vie privée dans 'Big Data'

2.2 Systèmes distribués

Ils peuvent être distingués selon le niveau de centralisation, ce qui permet de les classer en trois grandes catégories : centralisé, décentralisé et hybride, et ils peuvent être aussi

distingués selon le type de structuration : structuré et non structuré, comme illustré dans la figure 2.1.

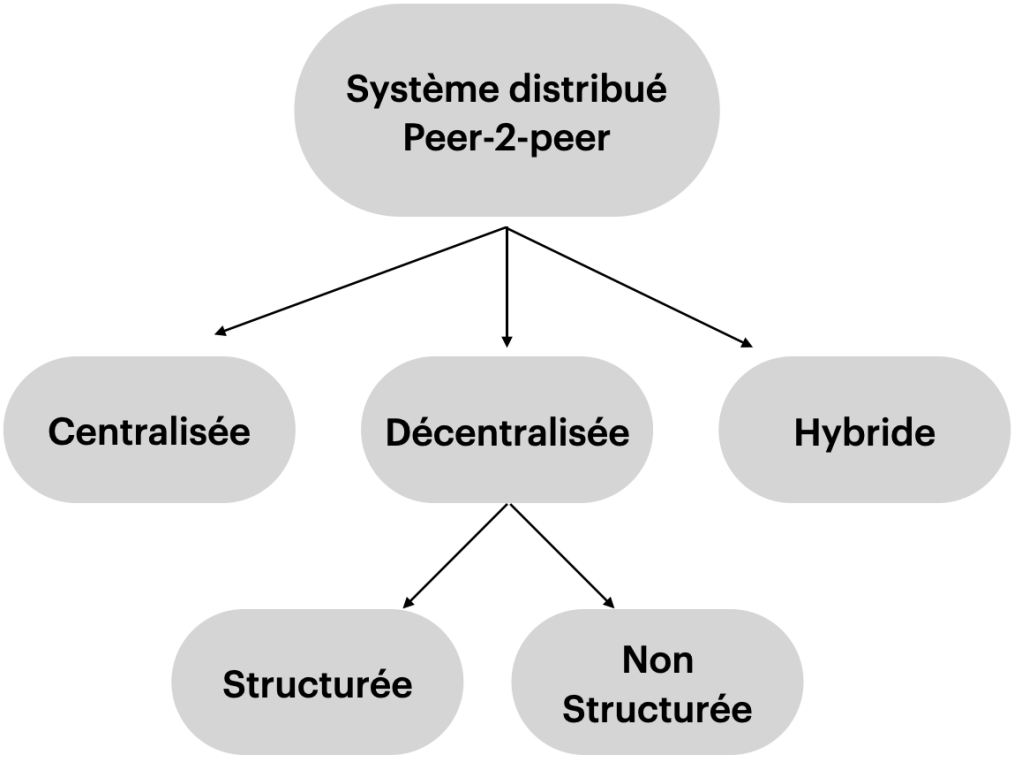


FIGURE 2.1 – Classification d'un système distribué P2P

2.2.1 Niveau de décentralisation

Un système peut être considéré comme peer-to-peer si au moins deux nœuds sont décentralisés. La mesure de degré de décentralisation des nœuds peut varier d'un système à un autre. D'après [4] tous les réseaux peer-to-peer ne sont pas complètement décentralisés. Nous faisons une distinction entre les topologies peer-to-peer centralisées, purement décentralisées et hybride.

Architecture centralisée

C'est la première architecture conçue pour le premier protocole P2P Napster [19].

Napster, contient un serveur central servant d'annuaire et qui exécute des fonctions vitales pour le système. Il contient les méta données sur les données partagées ainsi les informations sur les nœuds actifs tels que l'adresse IP, les services disponibles, les numéros de ports. Si un nœud est intéressé par une des ressources, il demande au serveur de trouver le pair pour échanger les données directement entre eux sans l'intermédiaire du fournisseur ce qui fait la distinction avec l'architecture client-serveur.

Les avantages de cette architecture sont :

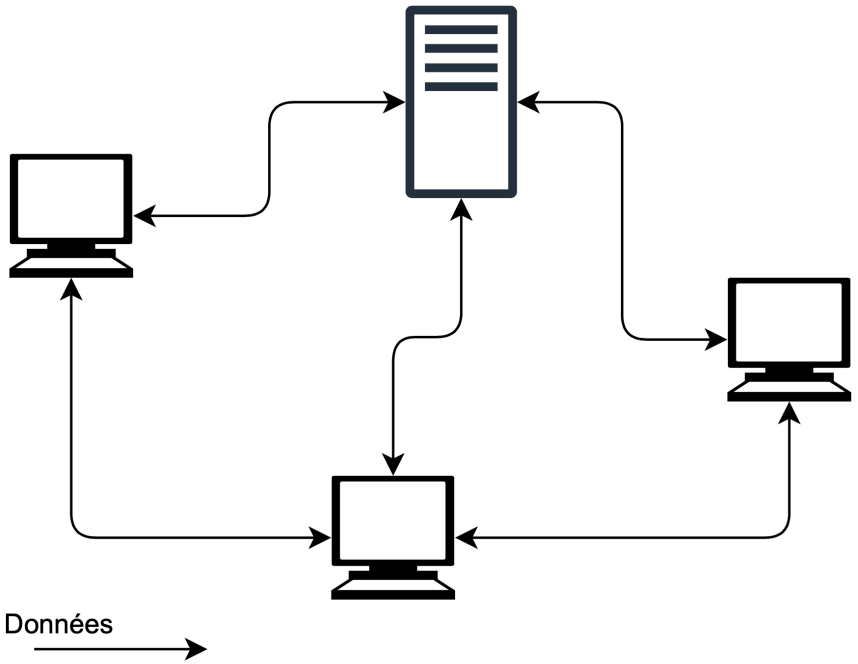


FIGURE 2.2 – Un système P2P centralisé

— Le serveur central facilite la recherche des données.
— La garantie d'échange des données.
— La réduction du trafic puisque seuls les noeuds contiennent des données.

Les inconvénients de cette architecture.

— - Chaque nœud doit s'identifier avec le serveur central ce qui ne garantit pas l'anonymat des nœuds.
— Si le serveur central ne peut être atteint l'ensemble du système cesse de fonctionner, ce qui impose une évolutivité limitée.

Architecture décentralisée

L'architecture décentralisée est appelée aussi l'architecture purement décentralisée où les nœuds échangent les données sans l'intervention d'un point centralisé.

Ce type d'architecture évite le seul point de défaillance avec un niveau élevé de tolérance aux pannes ce qui rajoute plus d'informations lors de la recherche des données. Freenet, et Gnutella sont des exemples de protocole P2P décentralisés.

Architecture hybride

L'architecture hybride est appelée aussi l'architecture partiellement centralisée.

Elle combine les deux architectures centralisées et décentralisées afin de bénéficier des avantages des deux. Dans les systèmes P2P hybrides, les paires ne sont pas égaux certaines paires ont plus de fonctions que les autres et jouent le rôle d'un responsable, ce qu'on appelle super Peer. Ce dernier est le serveur d'indexation pour une partie de réseau, il est élu selon sa disponibilité la bande passante et sa capacité de stockage. Cela réduit le

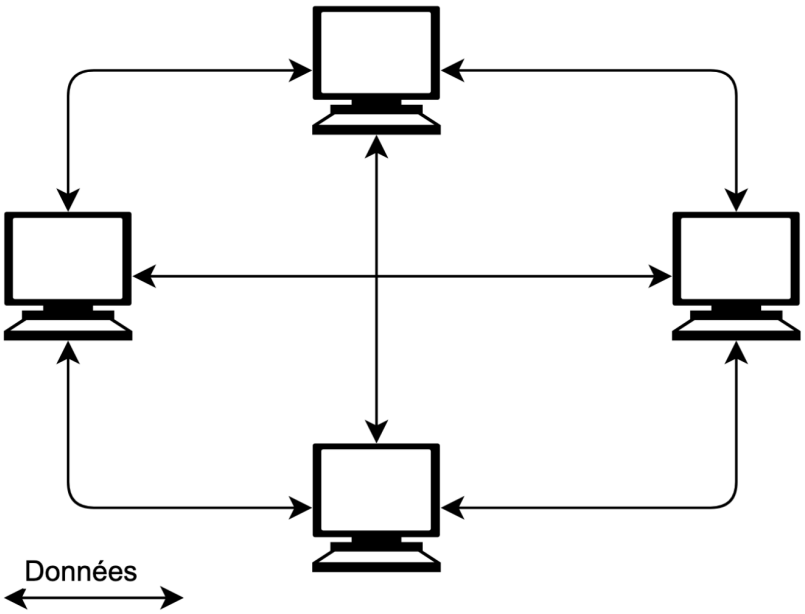


FIGURE 2.3 – Un système P2P décentralisé

retard de trouver l'emplacement d'une ressource demandé. Les risques de saturation et de point de défaillance sont très limités grâce à l'existence de plusieurs serveurs d'annuaire central les super nœuds. KaZaA est un protocole qui se repose sur cette architecture.

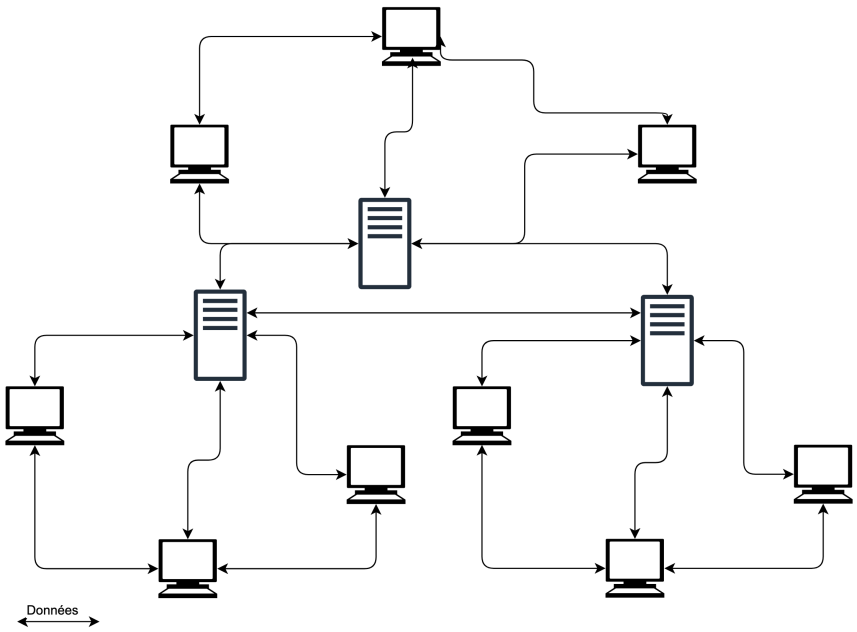


FIGURE 2.4 – Un système P2P hybride

2.2.2 Type de structuration

La structure du système dépend de la façon dont les nœuds et les données sont disposés dans le réseau.

Architecture structurée

Ce type de réseau résout les problèmes des architectures non-structurées, les pairs sont positionnés afin d'organiser le système et les nœuds peuvent localiser les ressources demandées. Afin d'éviter la propagation aléatoire des requêtes dans le système et effectuer une rechercher rapide et efficace un algorithme distribué de routage est mis en place. Des exemples de systèmes structurés sont Chord, CAN

Architecture non structurée

Un système est non structuré lorsque les nœuds n'ont pas de règles de positionnement spécifiques dans le réseau. L'emplacement des données ne suit aucune topologie. Chaque nœud ne possède que les informations sur ses ressources et aucune sur celles des autres, Il choisit les nœuds voisins de façon aléatoire. Ce système se caractérise par une forte consommation de la bande passante ce qui provoque l'arrêt d'une recherche d'une ressource avant d'atteindre le Peer possesseur ce qui implique une recherche lourde, inefficace qui entrave l'évolutivité. Gnutella est un protocole basé sur cette architecture.

2.2.3 Architecture décentralisée non structurée :

Dans un système purement décentralisé, tous les éléments jouent le rôle d'un client et serveur et des routeurs et des espaces de stockage, chaque nœud s'implique dans la recherche de données. Les paires ont une indexation que de leurs données. Pour attendre les ressources demandées il est impératif d'attendre autant de pairs que possible dans le système.

Si un nœud ou plusieurs nœuds se déconnectent du réseau suite à des pannes ou ont causé d'une défaillance le système continue à assurer le service.

Le défi majeur de ce système est de développer une méthode de recherche capable d'obtenir les résultats de recherche rapidement et efficace en présence des nœuds égaux.

Pour effectuer une recherche, les nœuds envoient des requêtes à leurs voisins en utilisant la méthode d'inondation, qu'ils transmettent eux-mêmes aux voisins, en limitant le nombre de sauts par un TTL (time-to-live), typiquement 7 pour Gnutella. Le propriétaire de la ressource répond alors au demandeur et échange de manière simple.

2.2.4 Architecture décentralisée structurée :

Dans un système décentralisé structuré les données et les nœuds sont mappés dans le but d'effectuer une recherche rapide et récupérer efficacement les ressources. Chaque nœud utilise un DHT : "Les tables de hachage distribuées fournissent une image complète des données distribuées à de nombreux nœuds, quel que soit leur emplacement réel." Pour fournir un routage efficace entre nœuds et assurer l'efficacité de la recherche.

Le DHT se base sur une fonction de hachage avec de bonnes propriétés, afin de définir un identifiant unique pour chaque nœud et chaque ressource dans le système. Un nœud

est identifié par un ID l'hash de son adresse publique et une chaque ressource est identifiée par un Key le hash de son nom.

Le but de diviser l'espace de clés en région et d'attribuer un noeud responsable pour chaque région en fonction de la proximité de l'identifiant et de la clé. Le DHT stocke ces informations sous une forme d'une paire (clé, valeur) dans tous les noeuds du système. De sorte que la défaillance d'un noeud ou plusieurs noeuds ne causera jamais la perte du réseau. Il permet de gérer un nombre illimité des entrées et gère aussi la connexion et le départ des noeuds dans le système.

2.3 La sécurité dans les systèmes P2P

Suite à la nature distribuée d'un système P2P, l'un des problèmes majeurs est de protéger les ressources partagées et assurer la continuité du service fourni. Afin de faciliter cette discussion et d'après [4] nous présentons d'abord les éléments fonctionnels d'un système P2P dans le but de relier les implications de sécurité pour des systèmes ou des cas d'application spécifiques. Par la suite une évaluation sur les risques des attaques afin d'atténuer les conséquences possibles.

Un système P2P doit protéger deux catégories fonctionnelles.

— *Les opérations P2P (OP-P2P)* : Ce sont les différentes opérations effectuées via l'interface de ce protocole P2P, par exemple : la recherche, téléchargement, la découverte, et le routage.

— *Les structures de données P2P (SD-P2P)* : Ce sont les fonctionnalités accessibles au niveau réseau ou directement sur le nœud, par exemple : les ressources à partager avec tous les éléments du système, les informations dans la table de routage dans les nœuds.

Dans la section suivante, nous discuterons des différentes attaques contre P2P, en utilisant les notions de sécurité [4, 20] la confidentialité, l'intégrité et la disponibilité, tout en se référèrent à OP-P2P et SD-P2P.

2.3.1 Les différents types d'attaques

Les attaques contre un système peer-to-peer cible les deux catégories OP-P2P et SD-P2P par la perturbation de la connexion et l'accès aux éléments du système ou par l'injection des données incorrectes. Par conséquence, on aura des pairs qui se trompe sur le routage, des ressources devenant injoignable, et la saturation du système.

Nous citons maintenant certaines attaques et l'impact correspondant sur la triade CIA (confidentialité, intégrité, disponibilité).

— *Les attaques par déni de service (DoS)* : C'est une attaque contre un nœud ou plusieurs nœuds dans un réseau qui consiste à provoquer la perte des ressources [4, 20]. L'attaque DoS devient plus destructive si plusieurs nœuds sont impliqués, on parle alors d'une attaque DDOS. Dans le cas des architecture P2P, l'attaquant essai d'inonder excessivement le réseau par des faux paquets empêchant (OP-P2P)

où il vise à éliminer un ensemble spécifique de pairs. Cela pourrait affecter le lien entre pairs et endommager les DHT (Distributed Hash Table) et envahir la bande passante ce qui entraîne un arrêt de service du protocole P2P.

— *Les attaques par collusion* [21] : L'attaquant collecte les différentes informations confidentielles des nœuds du système afin de menacer la disponibilité, l'intégrité des ressources partagés en exploitant l'injection de fausses données via un ou plusieurs nœuds compromis. Cette attaque affecte négativement l'OP-P2P.

— *Les attaques de pollution* [22, 23] : L'attaque de pollution ou l'empoisonnement de l'index, sont des attaques courantes dans le système P2P.Cela Consiste à collecter et remplacer les ressources par des faux afin de dégrader et de compromettre l'intégrité du service assuré par le système. Les auteurs [24] décrivent une stratégie de pollution en insérant des nœuds pollueurs et de les rendre responsables à l'indexation des ressources et générer directement des réponses corrompues.

— *Les attaques de censure* [23] : L'attaque de censure permet à un pair ayant une mauvaise réputation de s'échapper aux conséquences de l'abus en utilisant des vulnérabilités du système. Une fois qu'ils ont restauré leur réputation les attaquants continuent avec leur comportement malveillant. Cette attaque vise l'intégrité et la disponibilité des systèmes P2P basés sur la réputation. Elle met en danger la SD-P2P.

— *Les attaques 'Routing attack'* [25, 26] vise à compromettre la capacité du réseau à transmettre les ressources entre les nœuds. Un pair malveillant agit sur le mécanisme d'échange afin que les messages atteignent leurs destinations après le délai maximal ou pas du tout. Cette attaque compromet la disponibilité et l'intégrité du réseau et vise négativement la SD-P2P

— *Les attaques 'whitewashing'* [27] Dans la plupart des systèmes P2P, les utilisateurs ont la possibilité de changer leur ID à tout moment sans avoir de pénalités. Cette fonctionnalité aide les nouveaux entrants et le réseau à croitre rapidement. Dans le même temps, aucun utilisateur malveillant ne sera pénalisé pour avoir exploité cette vulnérabilité et détourner le système. Ce comportement est l'un des principaux problèmes pour ce système. Cette attaque met en danger le OP-P2P.

— *Les attaques Sybil* [28] Cela consiste à attaquer les systèmes basés sur la réputation par la création d'un nombre important de pairs malveillants avec de nombreuses identités volées afin d'attirer le maximum de trafic et de bâtir une très large influence sur le réseau. Le nombre d'identités générées dépent des ressources du pair attaquant telles que la puissance de calcul, la bande passante, avec l'évolution rapide de la technologie et la connexion internet à haut débit et le cloud l'attaquant peux causer des dommages considérables pour des grands protocoles P2P. Cette attaque compromet la disponibilité et la confidentialité et affecte la OP du système. Parmi les attaques importantes de Sybil, figure le compromis de la DHT.

— *Les attaques Eclipse* [29] : Vise principalement les systèmes P2P structurés. Dans ce type de réseau, chaque nœud est connecté par les nœuds les plus proches. L'attaque consiste à contrôler un grand nombre des nœuds voisins, une fois que l'attaquant a réalisé cela, il peut alors séparer le réseau en différents sous-réseaux, afin d'empêcher le trafic d'être redirigé vers le pair sain. Par conséquent, cela modifie l'utilisation normale du routage. Cette attaque a pour but de masquer les pairs sains sur le réseau

d'où le nom 'Eclipse'. Il s'agit d'une attaque composée qui implique Routing attack, Sybil déni de service, collusion whitewashing. Elle impacte les deux fonctionnalités OP-P2P et SD-P2P

Attaque	Disponibilité	Integrité	Confidentialité	Functionality
DoS/DDoS	✓	✘	✘	P-OP
Collusion	✓	✓	✓	P-OP
Pollution	✘	✓	✘	P-DS
White washing & censorship	✓	✓	✘	P-DS
Routing	✓	✓	✘	P-DS
Buffer map cheating	✓	✓	✘	P-OP
Sybil	✓	✘	✓	P-OP
Eclipse	✓	✓	✓	P-DS,P-OP

FIGURE 2.5 – Les attaques les objectifs de sécurité et les fonctionnalités affectées

Le tableau 2.5 [4,30] répertorie les différentes attaques sur les deux catégories OP-P2P et SD-P2P qui dégradent et compromettent les services. La complicité hostile de pairs malveillants est un facteur clé dans le déclenchement et le déroulement de ces attaques. Les options de conception inhérentes au P2P sont utilisées pour promouvoir l'évolutivité et la résilience. Les attaques contre les systèmes peer-to-peer affectent généralement la confidentialité, l'intégrité ou la disponibilité du système. Certaines des attaques observées sont connues d'autres architectures de système.
L'utilisation du P2P comporte de nombreux risques, des techniques de protections évoluent parallèlement à ces attaques. Ainsi, face à l'amélioration du filtrage et de l'identification des pairs, le P2P répréhensible, chiffré ou anonyme évolue et se développe pour former un P2P plus solide.

Les attaques par déni de service dégradent ou empêchent un système de corriger la prestation des services [31, 32]. L'attaque la plus sophistiquée de Sybil [32–34] peut être utilisée comme précurseur potentiel d'une attaque Eclipse [32, 33].
Si des mécanismes de stockage sécurisés, de routage sécurisé ou d'authentification ne peuvent pas être fournis, un ensemble d'attaques comprenant l'omission, la contrefaçon de contenu, la pollution de contenu, la censure ou l'empoisonnement de table de routage peuvent se produire [32, 34].

2.3.2 Les mécanismes de sécurité dans les systèmes P2P

La sécurité d'un système P2P se repose sur les principaux mécanismes suivants : l'authentification, un routage sécurité et un stockage sécurisé.

— *Les mécanismes d'authentification* [32, 35] : Ils permettent à un ensemble des paires de se connecter entre eux et à échanger les données et les stocker de façon sécurisée.

— *Le stockage sécurité* est primordial pour les systèmes P2P comme dans les systèmes centralisés, il empêchait les attaquants d'apporter des modifications illicites aux données [31, 33, 35, 36]. Par exemple dans les jeux en ligne, la modification illicite des données du jeu est considérée comme une tricherie. [34].

— Le routage sécurisé permet d'assurer la transmission des données aux pairs sains [33, 35, 36].

Les scénarios Sybil : se produit lorsque une attaque avec un petit nombre des pairs malveillants pour recueillir plusieurs informations sur l'ensemble du système. Ce qui permet a ces malveillants de simuler un plus grands nombre de pairs qu'au départ de l'attaque.

Les scénarios de routage : Les mécanismes de défense pour envisager les attaques de routage est d'attribuer plusieurs routes pour chaque recherche à l'aide des chemins disjoints, ce qui provoque plus de frais aux messages. L'autre alternative se repose sur le soutien des protocoles et des algorithmes cryptographiques pour assurer la sécurité des chemins. La coordination des signatures cryptographique à l'échelle du système décentralisé est difficile à réaliser.

Les mécanismes de sécurité cités au-dessus prouvent la résilience des systèmes P2P contre les différentes attaques. Naturellement, ces mécanismes ne sont résilients que jusqu'à ce qu'une masse critique de pair malveillants de collusion soit atteinte. En outre, certains de ces mécanismes nécessitent un soutien cryptographique ou l'identification des pairs. Ces exigences peuvent interférer avec les exigences d'application telles que l'anonymat, l'hétérogénéité ou la frugalité des ressources.

2.4 Big Data : solutions, sécurité et respect de la vie privée

Le Big Data [37, 38] se réfère spécifiquement à des ensembles de données volumineux et complexes que les applications de traitement des données traditionnelles ne sont pas suffisantes. Le Big Data s'agit du grand volume de données, structurées et non structurées, qui peut inonder une entreprise au quotidien. En raison du développement technologique récent, la quantité de données générées sur Internet via les réseaux sociaux, les réseaux de capteurs et les applications de santé, augmente considérablement de jour en jour. La quantité enorme de données produites à partir de diverses sources dans de multiples formats à très grande vitesse [39, 40] est appelée Big Data. Le terme Big Data est défini par [41] comme « une nouvelle de technologies et d'architectures, conçues pour séparer économiquement la valeur des très grands volumes d'une grande variété de données, en permettant le traitement, et l'analyse avec grande vitesse ».

Sur la base de cette définition, les propriétés du Big Data sont les 3V, Volume, Vitesse et Variété.

D'autres études ont souligné que la définition des 3Vs est insuffisante pour expliquer le Big Data à laquelle nous sommes confrontés aujourd'hui. Ainsi, la véracité, la validité, la valeur, la variabilité, ont été ajoutés pour apporter une explication complémentaire du Big Data [42]. La caractéristique principale du Big Data est que les données sont diverses, c'est-à-dire qu'elles peuvent contenir du texte, de l'audio, de l'image ou de la

vidéo, etc. Ces différents types de données sont définies par la variété. Afin d'assurer la protection de la vie privée des utilisateurs en Big Data, divers mécanismes ont été développés ces dernières années. Ces mécanismes peuvent être regroupés en fonction des étapes du cycle de vie du Big Data [43], comme illustré dans la figure 2.6, la génération, le stockage et le traitement des données. Dans la phase de génération de données, des techniques de contrôle d'accès sont utilisées pour la protection de la vie privée. Dans la phase de stockage, les techniques de protection de la vie privée se basent sur les protocoles et les algorithmes de chiffrement. Les techniques basées sur le chiffrement peuvent être divisées en chiffrement basé sur l'identité (IBE), en chiffrement basé sur les attributs (ABE) et en chiffrement des lieux de stockage. En outre, pour protéger les informations sensibles, des Cloud hybrides sont utilisés lorsque des données sensibles sont stockées dans des Cloud privés. La phase de traitement des données intègre la publication de données de préservation de la vie privée (PPDP) et l'extraction des connaissances à partir des données. Dans PPDP, des techniques d'anonymisation telles que la généralisation et la suppression sont utilisées pour protéger la confidentialité des données. Ces mécanismes peuvent être divisés en techniques de regroupement, de classification et d'association fondées sur l'exploitation minière. Bien que le regroupement et la classification divisent les données d'entrée en divers groupes, les techniques fondées sur l'exploration des règles d'association trouvent les relations et les tendances utiles dans les données d'entrée. [44]. Pour gérer diverses mesures du Big Data en termes de volume, de vitesse et de variété, il est nécessaire de concevoir des cadres efficaces pour traiter la mesure étendue des données arrivant à très grande vitesse à partir de diverses sources. Le Big Data doit connaître plusieurs phases au cours de son cycle de vie.

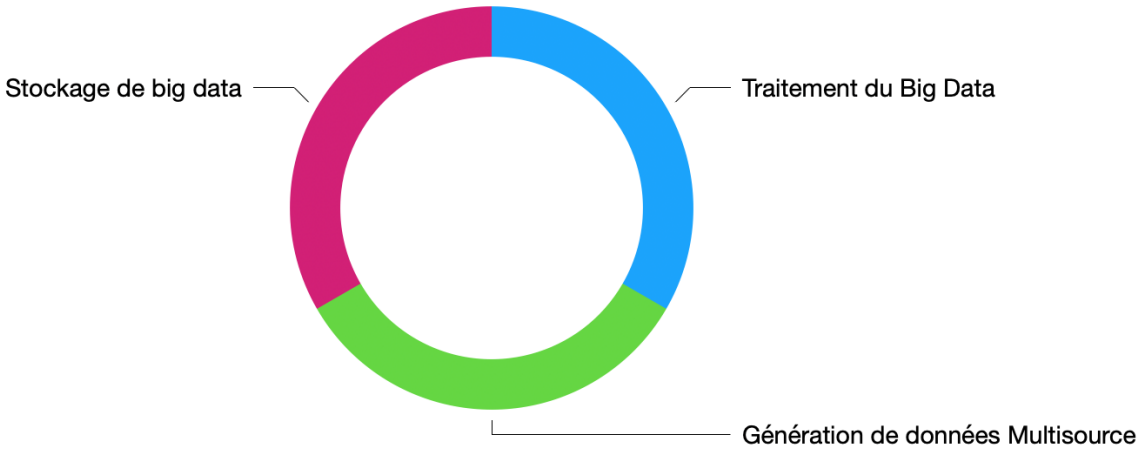


FIGURE 2.6 – Cycle de vie du Big Data

En 2012, 2,5 quintillions d'octets de données sont créés quotidiennement. Les volumes de données sont vastes, la vitesse de génération des données est rapide et l'espace données/informations est large [45].

Dans le monde numérique d'aujourd'hui, où beaucoup d'informations sont stockées dans le Big Data, l'analyse des bases de données peut fournir la possibilité de résoudre de

grands problèmes de la société comme les soins de santé ou autre. L'analyse intelligente du Big Data énergétique est également un sujet très complexe et stimulant qui partage de nombreux problèmes communs avec l'analyse générique du Big Data. Comme la base de données contient des renseignements personnels, elle est vulnérable pour fournir un accès direct aux attaquants et aux analystes. Des données sur la vie privée des individus peuvent être divulguées.

2.4.1 Défis de sécurité et la protection de la vie privée

La Sécurité

La sécurité consiste à protéger les informations des utilisateurs contre l'accès non autorisé, la perturbation, la modification, l'inspection, l'enregistrement et la destruction.

La protection de la vie privé (Privacy) : C'est le privilège d'avoir un certain contrôle sur la façon dont les informations sensibles sont obtenues et utilisées. La protection de la vie privé est le droit d'une personne ou d'un groupe d'éviter que des informations le concernant ne soient exposées à des personnes autres que celles qui ont droit. Une préoccupation importante concernant la protection de la vie privée des utilisateurs est la découverte des données personnelles via la transmission Internet [46].

	Respect de la vie privée	Sécurité
1	Le respect de la vie privée est l'utilisation appropriée des informations de l'utilisateur	La sécurité est la « confidentialité, l'intégrité et la disponibilité » des données
2	Le respect de la vie privée est la capacité de décider quels renseignements d'une personne	La sécurité offre la possibilité d'être sûr que les décisions sont respectées
3	La question de la protection de la vie privée s'applique souvent au droit du consommateur afin de protéger ses données	La sécurité peut assurer la confidentialité. L'objectif global de la plupart des systèmes de sécurité est de protéger une entreprise ou une agence
4	Il est possible d'avoir une mauvaise confidentialité et de bonnes pratiques de sécurité	Toutefois, il est difficile d'avoir une bonne confidentialité sans un bon programme de sécurité des données
5	Si l'utilisateur effectue un achat auprès de XYZ Company et fournit les informations de paiement et d'adresse afin qu'il puisse expédier le produit, il ne peut alors pas vendre les informations de l'utilisateur à un tiers sans le consentement préalable de l'utilisateur.	La société XYZ utilise diverses techniques (chiffrement, Pare-feu) afin d'empêcher le compromis de données des vulnérabilités dans le réseau

TABLE 2.1 – Les différences supplémentaires entre la protection de la vie privée et sécurité

2.4.2 Spécifications de la protection de la vie privée Big Data

L'analyse du Big Data est utilisée dans diverses organisations ; une grande partie d'entre eux choisit de ne pas utiliser ces services en raison du manque de mécanismes de protection de la vie privée.

Les sections suivantes analysent les stratégies possibles pour mettre à niveau les plates-formes de données volumineuses à l'aide des fonctionnalités de protection de la vie privée.

Les entreprises et les organismes gouvernementaux génèrent et collectent en continu de grandes quantités de données. L'attention est mise aujourd'hui sur la quantité substantielle de données créera sans aucun doute, des opportunités et des moyens de comprendre le traitement de ces données dans de nombreux domaines différents.

Mais il faut s'assurer que la conformité aux règles de la préservation de la vie privée dans d'analyse des données massives et d'exploration de données.

Les développeurs doivent être en mesure de vérifier que leurs applications sont conformes aux conventions du respect de la vie privée et que les informations sensibles sont confidentielles, quelles que soient les modifications apportées aux applications aux règles de confidentialité. Pour relever ces défis, il convient d'identifier la nécessité de nouvelles contributions dans les domaines des méthodes formelles et des procédures de test. De nouveaux paradigmes pour les tests de conformité du respect de la vie privée dans r les quatre zones du processus ETL (Extract, Transform, and Load) comme illustré à la figure 2.6 [47, 48].

1. *Validation de processus Pre-hadoop* : Cette étape fait la représentation du processus de chargement des données. À cette étape, les spécifications de confidentialité caractérisent les données sensibles qui peuvent identifier de manière unique un utilisateur ou une entité. Les conditions de confidentialité peuvent également indiquer les données pouvant être stockées et la durée de conservation. À cette étape, des restrictions de schéma peuvent également se produire.
2. *Validation de processus Map-Reduce* : Ce processus modifie les données volumineuses pour réagir efficacement à une requête. Les conditions de confidentialité peuvent indiquer le nombre minimal d'enregistrements renvoyés requis pour couvrir des valeurs individuelles, en plus des contraintes sur le partage de données entre différents processus.
3. *Validation de processus ETL* : Comme à l'étape (2), La justification de l'entreposage doit être confirmée à cette étape pour la conformité aux conditions de confidentialité. Certaines valeurs de données peuvent être agrégées de manière anonyme dans l'entrepôt si cela indique une probabilité élevée d'identification des individus. Les rapports concernent une autre forme de questions, avec une visibilité accrue et un public plus large. Les termes de confidentialité qui caractérisent les "objectifs" sont fondamentaux pour vérifier que les données sensibles ne sont pas signalées à l'exception des utilisations spécifiées.

2.4.3 Confidentialité de Big Data dans le processus de création de données

La génération de données peut être classée en génération de données actives et la génération de données passives. Par génération de données active, nous voulons dire que le propriétaire des données fournira les données à un tiers [49], tandis que la génération passive des données fait référence aux circonstances dans lesquelles les données sont produites par les actions en ligne du propriétaire des données (par exemple, la navigation) et que le propriétaire des données peut ne pas savoir que les données sont collectées par

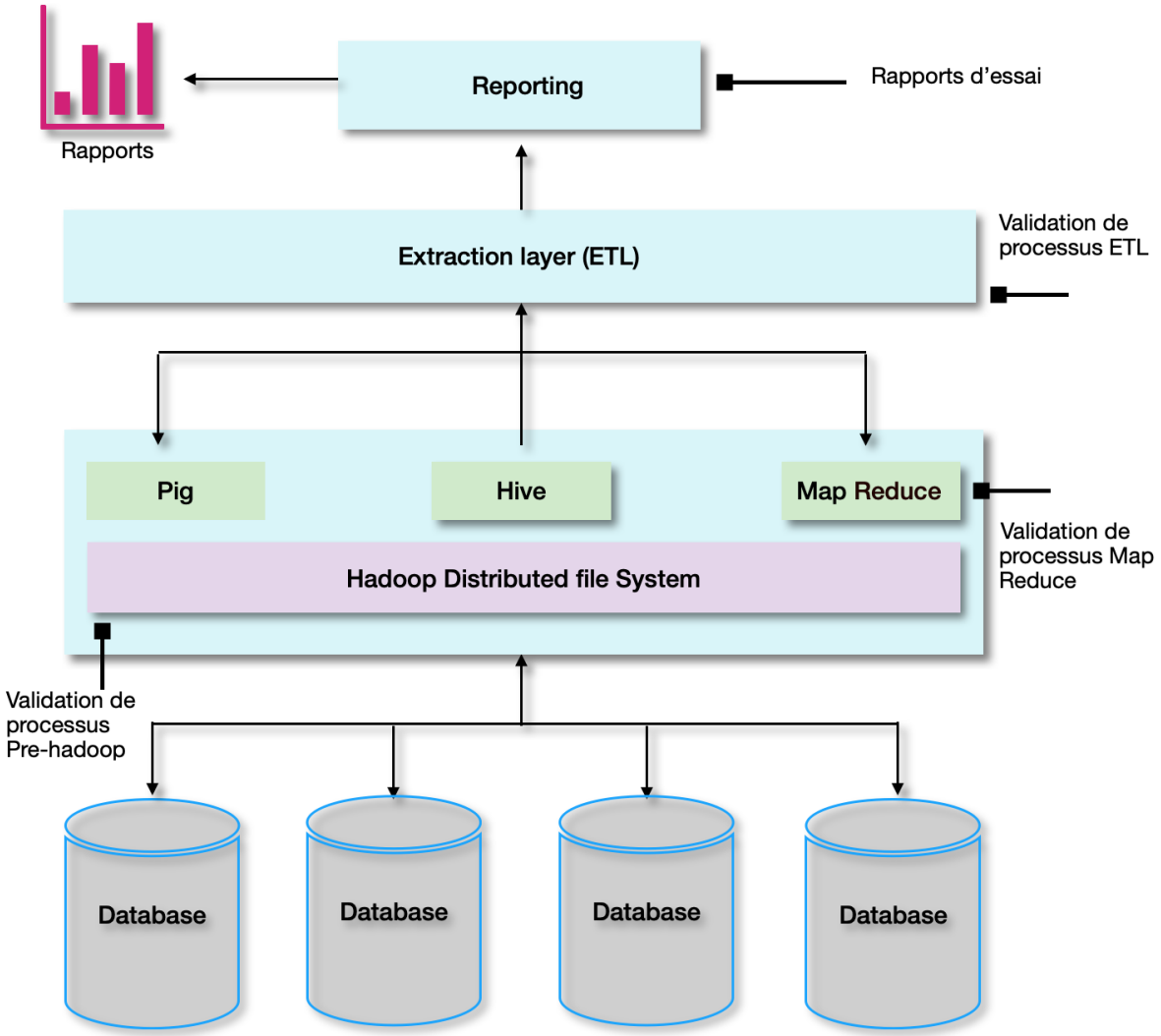


FIGURE 2.7 – Architecture de Big Data et zone de test de conformité et de la protection de la vie privée sur les quatre zones de l'ETL (Extraction, Transformation et Chargement)

un tiers. Réduction du risque de violation de la confidentialité lors de la génération de données, soit en limitant l'accès, soit en falsifiant les données.

1. *Restriction d'accès* : Si le propriétaire des données estime que les données peuvent révéler des informations sensibles qui ne sont pas censées être partagées, il refuse de fournir ces données. Si le propriétaire des données donne les données passivement, quelques mesures peuvent être prises pour assurer la confidentialité, telle que des extensions anti-suivi, des bloqueurs de publicités ou des scripts et des outils de chiffrement.
2. *Falsification de données* : Dans certaines circonstances, il est irréaliste de contrecarrer l'accès aux données sensibles. Dans ce cas, les données peuvent être déformées à l'aide de certains outils avant que les données ne soient récupérées par un tiers. Si les données sont déformées, les véritables informations ne peuvent pas être facilement révélées. Les techniques suivantes sont utilisées par le propriétaire des données pour cacher les données :

— Un outil Socketpuppet peut etre utilisé pour cacher l'identité en ligne d'un individu par la tromperie. En utilisant plusieurs Socketpuppets, les données appartenant à une personne spécifique seront considérées comme ayant une place avec différentes personnes. De cette façon, le collecteur de données n'aura pas suffisamment de connaissances pour relier différents socketpuppets à un seul individu.

— Certains outils de sécurité peuvent être utilisés pour masquer l'identité de l'individu, comme Mask Me. Ceci est particulièrement utile lorsque le propriétaire de données doit donner les détails de la carte de crédit au milieu des achats en ligne.

2.4.4 Confidentialité du Big Data en phase de stockage de données

Le stockage des données à fort volume n'est pas un défi majeur en raison de l'avancement des technologies de l'ère des données, par exemple, l'essor de l'informatique en Cloud [50]. Si le système de stockage de Big Data est compromis, il peut être exceptionnellement destructeur car les renseignements personnels des individus peuvent être divulgués [51]. Dans un environnement distribué, une application peut avoir besoin de plusieurs ensembles de données provenant de divers centres de données et donc relever le défi de la protection de la confidentialité.

Les mécanismes de sécurité conventionnels pour protéger les données peuvent être divisés en quatre catégories. Il s'agit des systèmes de sécurité des fichiers, des systèmes de sécurité des bases de données, des systèmes de sécurité des médias et des systèmes de cryptage des applications [52]. En réponse à la nature 3V de l'analyse du Big Data, l'infrastructure de stockage doit être évolutive. Il doit avoir la possibilité d'être configuré dynamiquement pour accueillir diverses applications. Une technologie prometteuse pour répondre à ces exigences est la virtualisation du stockage, habilitée par le paradigme émergent de l'informatique en Cloud [53]. La virtualisation de stockage est un processus dans lequel de nombreux périphériques de stockage réseau sont combinés dans ce qui donne l'impression d'être un seul périphérique de stockage. SecCloud est l'un des modèles de sécurité des données dans le Cloud qui considère conjointement la sécurité du stockage de données et la sécurité de l'audit de calcul dans le Cloud [54]. Par conséquent, il y a une discussion limitée en cas de confidentialité des données lorsqu'elles sont stockées sur le Cloud.

Approches de la protection de la confidentialité des données dans le Cloud

Lorsque les données sont stockées sur le cloud, la sécurité des données a principalement trois dimensions : la confidentialité, l'intégrité et la disponibilité [55]. Les deux premiers sont directement liés à la confidentialité des données, c'est-à-dire que si la confidentialité ou l'intégrité des données est violée, cela aura un effet direct sur la vie privée des utilisateurs. La disponibilité de l'information vise à s'assurer que les parties autorisées sont en mesure d'accéder à l'information au besoin. Une exigence de base pour le système de stockage de Big Data est de protéger la vie privée d'une personne. Il existe certains mécanismes pour satisfaire à cette exigence.

Verification de l'intégrité des données

Du moment que les données sont stockées dans le Cloud, le propriétaire de données perd le contrôle sur ces données. En effet, les données externalisées sont à risque, car le serveur Cloud n'est peut-être pas entièrement fiable. Le propriétaire des données doit être fermement convaincu que le Cloud stocke correctement les données selon le contrat de niveau de service. Assurer la confidentialité de l'utilisateur du Cloud est de fournir au système le mécanisme permettant au propriétaire de données de vérifier que ses données, stockées sur le Cloud, sont intactes [56, 57]. L'intégrité du stockage de données dans les systèmes traditionnels peut être vérifiée par plusieurs façons : le code Reed-Solomon, les checksums, les fonctions de hachage trapdoor, le code d'authentification des messages (MAC), les signatures numériques, etc. Par conséquent, la vérification de l'intégrité des données est d'une importance cruciale. Il compare les différents systèmes de vérification de l'intégrité discutés [56, 58]. Pour vérifier l'intégrité des données stockées sur le Cloud, l'approche simple consiste à récupérer toutes les données du Cloud. Vérifier l'intégrité des données sans avoir à récupérer les données du Cloud [57,58]. Dans le cadre du système de vérification de l'intégrité, le serveur Cloud ne peut fournir les preuves substantielles d'intégrité des données que lorsque toutes les données sont intactes. Il est fortement prescrit que la vérification de l'intégrité devrait être effectuée régulièrement afin d'assurer le plus haut niveau de protection des données [58].

D'une façon générale, les systèmes de traitement Big Data sont classés en trois lots : flux, graphiques et apprentissage automatique [59,60]. L'objectif de la confidentialité est de protéger l'information contre la divulgation non autorisée puisque les données recueillies peuvent contenir des informations sensibles du propriétaire des données. Un deuxième objectif peut être d'extraire des informations significatives des données sans violer la confidentialité.

2.4.5 La protection de la vie privée dans le Big Data

Peu de méthodes traditionnelles de préservation de la confidentialité dans les données massives sont décrites brièvement ici. Ces méthodes sont traditionnellement utilisées pour assurer une certaine confidentialité, mais leurs démérites ont conduit à l'avènement de nouvelles méthodes.

La dé-identification [61, 62] est une technique traditionnelle d'exploration de données et de préservation de la vie privée par la généralisation (remplacement des quasi-identificateurs par des valeurs moins particulières, mais sémantiquement cohérentes) et la suppression (ne libérant pas du tout certaines valeurs) avant la publication. Afin d'atténuer les menaces de ré-identification ; les concepts de k-anonymat [61, 63], l-diversité [62,64] et t-closeness [61,64] ont été introduits pour améliorer l'exploration traditionnelle des données de protection de la vie privée.

K-anonymat

On dit qu'une diffusion des données à la propriété de K-anonymat [61,65], si les identifiants pour chaque utilisateur contenu dans la publication ne peuvent être perçus d'au moins $k-1$ personnes dont les renseignements apparaissent dans la publication. Dans le

contexte des problèmes de k-anonymisation, une base de données est une table qui se compose de n lignes et de m colonnes, chaque ligne de la table représentant un enregistrement relatif à un individu particulier d'une population et les entrées dans les différentes lignes ne doivent pas être uniques. Les valeurs dans les différentes colonnes sont les valeurs des attributs liés aux membres de la population.

Introduit par Sweeny dans son papier [2] en 2002, le K-Anonimity fait appel à deux notions de base, les attributs et les quasi-identifiers définis ci-dessous :

Definition 1. Les attributs [2] : On considère le tableau $B(A_1, ...A_n)$ avec un nombre fini des tuples. L'ensemble fini des attributs de B sont notés $\{A1, \ldots, An\}$. Soit le tableau $B(A_1, \ldots, A_n)$, $\{A_i, \ldots, A_j\} \subseteq \{A_1, \ldots, A_n\}$, et un tuple $t \in B$, on utilise $t\,[A_i, \ldots, A_j]$ pour designer la séquence des valeurs,$v_i, \ldots, v_j, deA_i, \ldots, A_j$ dans t.

Definition 2. Quasi-identifier [2] : Pour un ensemble d'entités, $T(A_1, \ldots, A_n)$, $f_c : U \rightarrow$ T et $f_g : T \rightarrow U'$, avec $U \subseteq U'$. A quasi identifier de T, s'ecrit sous la forme QT, est un ensemble d'attributs $\{A_i, \ldots, A_j\} \subseteq A_1, \ldots, A_n$, il $\exists p_i \in U$ telque $f_g(f_c(p_i)\,[QT]) = p_i$.

Definition 3. k-anonymity [2] : Pour définir le k-anonymity, on considère $RT(A_1, ..., A_n)$ une table et QI_{RT} son quasi-identifier. RT satisfait le k-anonymity si et seulement si pour chaque valeur dans $RT\,[QI_{RT}]$ apparait au moins k fois dans $RT\,[QI_{RT}]$.

À titre d'exemple, le tableau 2.2 est une base de données non anonymisée comprenant les dossiers des patients de certains hôpitaux fictifs d'Hyderabad. Il y a six attributs ainsi que dix enregistrements dans ces données. Il y a deux techniques régulières pour réaliser le k-anonymat pour une certaine valeur de k.

Nom	Age	Sex	Adresse	Religion	Maladie
Roch	19	Femelle	Paris	Judaïsme	Cancer
Adrien	35	Male	Grenoble	Christianisme	Dermatologique
Geoffroy	29	Male	Toulouse	Christianisme	Respiratoire
Phil	27	Male	Strasbourg	Christianisme	Respiratoire
Émilien	50	Male	Paris	mithraïsme	Cancer
Sacha	10	Femelle	Paris	Judaïsme	Génétique
Théodore	41	Male	Montpellier	Judaïsme	Cancer
Odette	33	Femelle	Grenoble	Hindou	Génétique
Xavier	52	Male	Grenoble	Christianisme	Respiratoire
Sylvaine	10	Femelle	Paris	Christianisme	Respiratoire

TABLE 2.2 – Une base de données non anonyme comprenant les dossiers des patients d'un hôpital fictif

Nom	Age	Sex	Adresse	Religion	Maladie
*	$20 \leq Age \leq 30$	Femelle	Paris	*	Cancer
*	$30 \leq Age \leq 40$	Male	Grenoble	*	Dermatologique
*	$20 \leq Age \leq 30$	Male	Toulouse	*	Respiratoire
*	$20 \leq Age \leq 30$	Male	Strasbourg	*	Respiratoire
*	$50 \leq Age \leq 60$	Male	Paris	*	Cancer
*	$Age \leq 20$	Femelle	Paris	*	Génétique
*	$40 \leq Age \leq 50$	Male	Montpellier	*	Cancer
*	$30 \leq Age \leq 40$	Femelle	Grenoble	*	Génétique
*	$50 \leq Age \leq 60$	Male	Grenoble	*	Respiratoire
*	$Age \leq 20$	Femelle	Paris	*	Respiratoire

TABLE 2.3 – Tableau à 2-Anonymat

Le tableau 2.3 est de 2-anonyme en ce qui concerne les attributs «Age», «Sexe» et «État de domicile», car pour tout mélange de ces attributs se trouvant dans une ligne du tableau, il n'y a toujours pas moins de deux lignes avec ces attributs exacts. Les attributs qui sont disponibles pour un adversaire sont appelés "quasi-identifiants". Chaque tuple "quasi-identifiant" se trouve dans au moins k enregistrements pour un ensemble de données avec k-anonymity. Les données k-anonymes peuvent toujours être impuissantes contre des attaques telles que des attaques de correspondance non triées, des attaques temporelles et des attaques de versions complémentaires [2, 64].La complexité est de rendre les relations des entrées privées k-anonymes, tout en minimisant la quantité d'informations qui ne sont pas divulguées et en même temps assurer l'anonymat des individus jusqu'à un groupe de taille k, et retenir une quantité minimale d'informations pour atteindre ce niveau de confidentialité et de le transformer en problème d'optimisation $NP - hard$.

L-diversity

Il s'agit d'une forme d'anonymisation basée sur un groupe utilisé pour protéger la vie privée dans l'ensembles des données en réduisant la granularité de la représentation des données. Le modèle l-Diversity (Distinct, Entropy, Recursive) [2, 61, 65] est une extension du modèle k-anonymity qui diminue la granularité de la représentation des données à l'aide des méthodes telles que la généralisation et la suppression d'un enregistrement donné, qui est mappé sur au moins k enregistrements différents dans les données. Une instantiation du l-Diversity fait appel à son entropie [62] :

Definition.1 : (Entropie de l-Diversity) [62] La A table est un Entropie l-Diverse si, pour tout $q* -$block,

$\sum_{s \in S}(p_(q*, s)log(p_(q*, s')) \geq log(l)$, avec $p_(q*, s) = n(q*, s)/\sum_{s' \in S} n_(q*, s')$ est une fraction des tuples dans q*-block avec des attributs sensibles de valeurs s. Par conséquence, chaque q* -block a au moins l valeurs distincts pour les attributs sensibles.

Le modèle l-diversity gère quelques faiblesses du modèle k-anonymity dans lequel les identités protégées au niveau des k-individus ne sont pas égales à la protection des valeurs

sensibles correspondantes qui ont été généralisées ou supprimées, en particulier lorsque les valeurs sensibles d'un groupe présentent une homogénéité. Le modèle de l-diversité inclut la promotion de la diversité intragroupe pour les valeurs sensibles dans le mécanisme d'anonymisation. Si vous souhaitez rendre les données l-diverses, des données fictives sont insérées. Ces données fictives améliorent la preservation de la vie privée (privacy), mais pourraient entraîner des problèmes au cours de l'analyse. De plus, la méthode de l-diversity est sujette à des attaques [2] qui n'empêchent pas la divulgation d'attributs.

T-proximité/t-Closeness

Il s'agit d'une nouvelle amélioration de l'anonymisation de la l-diversité basée sur les groupes et utilisée pour préserver la vie privée dans l'ensemble des données en diminuant sa granularité de représentation. Cette réduction constitue un compromis qui entraîne une certaine perte d'adéquation entre la gestion des données et ses algorithmes d'exploration. Le modèle de t-Closeness [2, 61] étend le modèle de l-diversité en traitant distinctement les valeurs d'un attribut en tenant compte de la distribution des valeurs des données pour cet attribut.

Définition : Le principe de t-proximité Une classe d'équivalence est dite de proximité t si la distance entre la distribution d'un attribut sensible dans cette classe et la distribution de l'attribut dans la table entière n'est pas supérieure à un seuil t. on dit qu'une table a une t-proximité si toutes les classes d'équivalence ont une t-proximité . [66]

Une table est dite à t-proximité si toutes les classes d'équivalence ont une t-proximité. Le principal avantage de la t-proximité est qu'elle intercepte la divulgation d'attributs. Le problème de la t-proximité, c'est que plus la taille et la variété des données augmentent, plus les chances de ré-identification augmentent. L'approche brute-force qui examine chaque partition possible de la table pour trouver la solution optimale de la forme :
$n^{O(n))}m^{O(1)} = 2^{O(nlogn)}m^{O(1)}$.

Étude comparative des approches de de-identification de la protection de la vie privée

L'analyse avancée des données peut extraire des informations précieuses des données massives, mais elle présente en même temps un risque important pour la protection de la vie privée des utilisateurs [63]. Il existe de nombreuses approches proposées pour préserver la vie privée avant, pendant et après le processus d'analyse des données massives. Il y a eu de nombreuses approches proposées pour préserver la vie privée avant, pendant et après le processus d'analyse sur les données volumineuses. Ce Chapitre traite trois méthodes de protection de la vie privée , telles que le K-anonymat, le L-diversité et le T-proximité. Comme les données des consommateurs continuent de croître rapidement et que les technologies s'améliorent sans cesse, le compromis entre la violation et la préservation de la vie privée s'avérera plus intense. Le tableau 2.4 présente les mesures existantes d anonymisation qui préservent la vie privée dans les données et leurs limites pour les données massives.

	Techniques	Definitions	Limites	Complexité
1	K-anonymat	Il s'agit d'un framework pour la construction et l'évaluation d'algorithmes et de systèmes qui libèrent des informations ce qui peut être révélé sur les propriétés des entités qui doivent être protégées	connaissance du back-ground	$O(k \log k)$
2	L-diversité	On dit qu'une classe d'équivalence a l-diversité s'il y a au moins une valeur pour sensible. On dit qu'un tableau a la l-diversité si chaque classe d'équivalence de la table a la diversité L	La l-diversité peut être difficile et insuffisante pour empêcher la divulgation des attributs	$O(n^2/k)$
3	T-proximité	On dit qu'une classe d'équivalence a une t-proximité si la distance entre la distribution d'un attribut sensible dans cette classe et la distribution de l'attribut dans l'ensemble du tableau n'est rien de plus qu'un seuil t. Une table est dite t-proximité si toutes les classes d'équivalence ont t–proximité	La t-proximité exige que la distribution d'un attribut sensible dans n'importe quelle classe d'équivalence soit proche de la distribution d'un attribut sensible dans le tableau global	$2^{O(n)O(m)}$

TABLE 2.4 – Les mesures existantes de dés-identification préservant la protection de la vie privée et ses limites dans le Big Data ainsi que leurs complexités informatiques

HybridEx

Le modèle d'exécution hybride [67] est un modèle de protection de la vie privée dans le Cloud Computing. Il tient compte de la sensibilité des données avant l'exécution d'une tâche. Les quatre catégories dans lesquelles HybrEx MapReduce permet de créer de nouveaux types d'applications qui utilisent des Cloud publiques et privés sont les suivantes :

1. Map hybride. : La phase map est exécutée à la fois dans les clouds publics et privés, tandis que la phase de réduction n'est exécutée que sur un des clouds, comme le montre la figure 2.8A.
2. Partitionnement vertical : Il est illustré dans la figure 2.8B, les tâches de mappage et de réduction sont exécutées dans le cloud public à l'aide de données publiques comme entrée, mélangent les données intermédiaires entre elles et stockent le résultat dans le Cloud public. Le même traitement est effectué dans le Cloud privé avec des

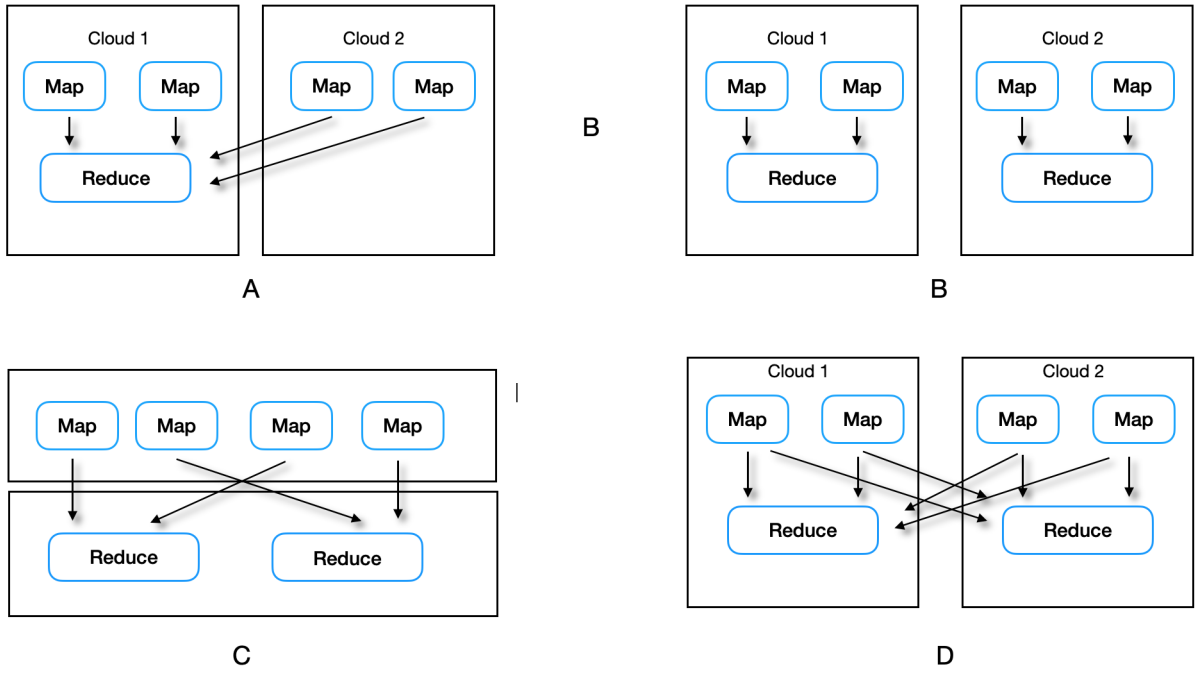


FIGURE 2.8 – Méthodes HybrEx : A Map Hybrid, B Partitionnement vertical, C Partitionnement horizontal et D Hybride, les quatre catégories dans lesquelles HybrEx MapReduce permet d'accéder à de nouveaux types d'applications utilisant à la fois des cloud publiques et privés.

données privées.

3. Partitionnement horizontal : la phase de mappage est exécutée sur les Clouds publics uniquement pendant que la phase de réduction est exécutée sur un Cloud privé, comme on peut le voir dans la figure 2.8C.
4. Comme illustré dans la figure 2.8D, la phase de mappage et la phase de réduction sont exécutées sur les Clouds publics et privés.

Des modèles de vérification d'intégrité complète et rapide sont également suggérés. Le problème avec HybridEx est qu'il ne traite pas la gestion des clés générées dans les Clouds.

2.4.6 Agrégation préservant la vie privée

L'agrégation de préservation de la vie privée [68] repose sur un chiffrement homomorphique utilisé comme technique de collecte de données populaires pour les statistiques d'événements. Avec un algorithme de chiffrement homomorphique, différentes sources peuvent utiliser la même clé publique pour chiffrer leurs données[39]. Ces textes chiffrés peuvent être agrégés et le résultat agrégé peut être récupéré avec la clé privée correspondante. Ainsi, l'agrégation qui préserve la vie privée peut protéger la confidentialité individuelle dans les phases de collecte et de stockage des données massives. En raison de sa flexibilité, il ne peut pas exécuter l'exploration de données complexes pour exploiter de nouvelles connaissances.

2.4.7 Opérations sur des données chiffrées

D'après [68], les opérations peuvent être exécutées sur des données chiffrées afin de protéger la vie privée individuelle dans l'analyse de données volumineuses. Étant donné que les opérations sur des données chiffrées sont pour la plupart complexes, qu'elles prennent du temps et que les données sont volumineuses et qu'elles ont besoin de nous pour extraire de nouvelles connaissances dans un délai raisonnable, l'exécution d'opérations sur des données chiffrées peut être qualifiée d'inefficace dans le cas de l'analyse des données massives.

2.4.8 Approches actuelles de préservation de la vie privée dans le Big Data

La confidentialité différentielle

Le differential privacy est une technique introduite par C. Dwork [69] qui permet d'obtenir des informations utiles aprtir des bases de données qui contiennent des informations sur les personnes sans révéler leurs identités personnelles. Pour ce faire, on introduit un minimum de distraction dans les informations fournies par le système de base de données. La distraction introduite est assez grande pour qu'ils protège la protection de la vie privée et en même temps assez petite pour que l'information fournie à l'analyste soit encore utile. Certaines techniques ont été utilisées précédemment pour protéger la confidentialité, mais elles se sont avérées infructueuses. Le mecanisme utilisé dans la figure 2.9 assure 1-differential privacy pour la requête *count*()

Définition : *Une Fonction aléatoire* κ *assure* ϵ *differential privacy si pour tous* $\mathcal{D}_1, \mathcal{D}_2$ *différent d'au plus une entrée, et tous* $\mathfrak{S} \subseteq$ *distance*(κ)

$$\frac{Pr\left[\kappa(\mathcal{D}_1) = \mathfrak{S}\right]}{Pr\left[\kappa(\mathcal{D}_2) = \mathfrak{S}\right]}$$

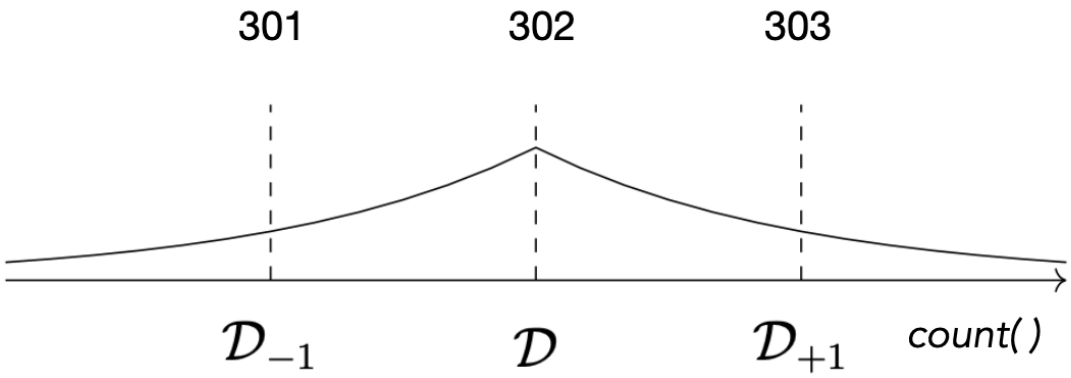


FIGURE 2.9 – Illustration de l'approche differential privacy avec un bruit laplacien $\varepsilon = 1$

L'artile de C. Dwork a également proposé un mécanisme basé sur l'ajout de bruit

laplacien au résultat d'une requête. La figure 2.9 montre ce mécanisme confidentialité différentielle.

L'ajout de ce bruit consiste à sélectionner un nombre aléatoire dans la distribution laplacienne afin de l'ajouter au résultat d'une requête. Dans l'exemple, la distribution de Laplace est représentée par une courbe centrée sur le dataset $\mathcal{D}$. On suppose qu'une requête nommé *count*() demande un calcul de cardinal de dataset $\mathcal{D}$. Le dataset $\mathcal{D}_{-1}$ représente un dataset voisin de $\mathcal{D}$ avec un enregistrement inférieur et $\mathcal{D}_{+1}$ un voisin avec un enregistrement de plus. Soit $|\mathcal{D}|$ égale à 300, le bruit laplacien est centré sur 300, un nombre aleatoire est séléctionné dans cette distribtuition suite à la requête. Une distribution laplacienne avec $\frac{1}{\varepsilon}$ comme un paramètre d'échelle, nous accorde une protection de la vie privée avec 1-differential privacy. Si ε est petite alors on aura une meilleure preservation de la vie privée. Mais aggraverait la qualité du résultat. Si on prend une autre requete autre que la fonction *count*() avec le meme bruit laplacien on n'aura pas la meme securité. Ceci est dû à la sensibilité de la fonction

Au milieu des années 90, la Commonwealth of Massachusetts Group Insurance Commission (GIC) a publié le dossier de santé anonyme de ses clients pour la recherche au profit de la société. Le GIC a masqué certaines informations telles que le nom, l'adresse postale, etc., afin de protéger la vie privée des utilisateurs. Latanya Sweeney (doctorante au MIT) a utilisé la base de données sur les électeurs et la base de données accessibles au public et publiées par le GIC, et a réussi à identifier le dossier de santé simplement en les comparant et en les associant. Ainsi, cacher certaines informations ne peut pas assurer la protection de l'identité individuelle.

Le differential privacy (DP) permet de résoudre ce problème comme le montre la figure 2.10. Dans DP, les analystes ne disposent pas d'un accès direct à la base de données contenant des informations personnelles. Un logiciel intermédiaire est introduit entre la base de données et l'analyste pour protéger la confidentialité. Ce logiciel intermédiaire est également appelé "protection de la vie privée".

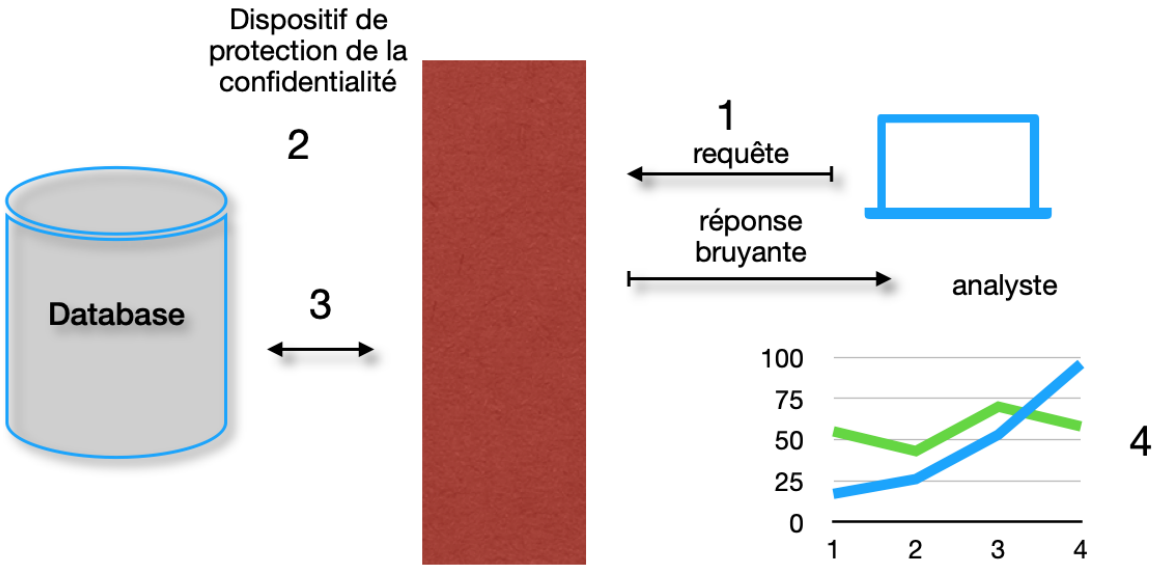


FIGURE 2.10 – Le differential privacy pour les données Big Data

— Etape1 : L'analyste peut effectuer une requête auprès de la base de données via ce dispositif de protection de la vie privée intermédiaire.
— Etape2 : Le dispositif de protection de la vie privée prend la requête de l'analyste et évalue cette requête et d'autres requêtes antérieures pour déterminer le risque sur la vie privée.
— Etape3 : Le dispositif de protection de la vie privée obtiendra alors la réponse de la base de données.
— Etape4 : consiste à ajouter une distorsion en fonction du risque évalué de protection de la vie privée et de le fournir à l'analyste.

Algorithme de protection de la vie privée dans le framework MapReduce

Dans la figure 2.11, le fournisseur de services ajoute un élément factice comme le bruit aux données de transaction originales recueillies par le fournisseur de données. Par la suite, un code unique est attribué aux éléments originaux. Le fournisseur de services maintient les informations de code pour filtrer l'élément factice après l'extraction d'éléments fréquents définis par une plate-forme Cloud externe. L'algorithme Apriori est effectué par la plate-forme Cloud externe à l'aide de données envoyées par le fournisseur de services. La plate-forme cloud externe renvoie l'ensemble d'éléments fréquents et la valeur de support au fournisseur de services. Le fournisseur de services filtre l'ensemble d'éléments fréquents qui est affecté par l'élément factice à l'aide d'un code pour extraire la règle d'association correcte en utilisant l'ensemble d'éléments fréquents sans l'élément factice.

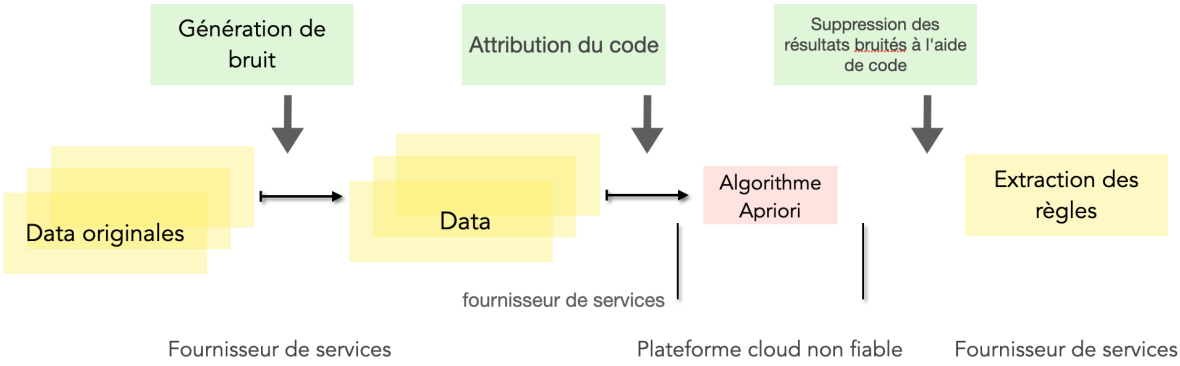


FIGURE 2.11 – Vue d'ensemble du processus d'exploration des règles d'association, le fournisseur de services ajoute un élément fictif en tant que bruit aux données de transaction d'origine collectées par le fournisseur de données

Protection de la vie privée lors de la publication de données volumineuses

La publication et la diffusion de données brutes sont des éléments cruciaux dans les applications commerciales, académiques et médicales avec un nombre croissant de plates-formes ouvertes, telles que les réseaux sociaux et les appareils mobiles à partir de laquelle les données pourraient être recueillies. Le volume de ces données a également augmenté au fil du temps [70]. Une grande partie du travail dans le domaine de la protection de la

confidentialité a porté sur la qualité de la protection de la vie privée et sur l'utilité des données publiées. La solution est de diviser les données en petits fragments et d'anonymiser chaque partie indépendamment [71].

Bien que le k-anonymat puisse empêcher les attaques d'identité, il ne parvient pas à protéger les données contre les attaques de divulgation d'attributs en raison du manque de diversité dans l'attribut sensible au sein de la classe d'équivalence. Le modèle l-Diversity exige que chaque classe d'équivalence ait au moins l valeurs sensibles bien représentées. Il est courant que des ensembles de données volumineux soient traités avec des plates-formes distribuées telles que le framework MapReduce [72, 73] afin de répartir un processus coûteux entre plusieurs noeuds et d'améliorer considérablement les performances. Par conséquent, afin de résoudre l'inefficacité, des améliorations des modèles de protection de la confidentialité sont introduites.

L'évaluation de la confiance joue un rôle important dans la gestion des fiducies. Il s'agit d'une approche technique de représentation de la confiance pour le traitement numérique, dans laquelle les facteurs influençant la confiance sont évalués sur la base de données probantes pour obtenir un nombre continu ou discret. Il propose deux régimes visant à préserver la confidentialité dans l'évaluation des fiducies. Afin de réduire les coûts de communication et de calcul, on propose d'introduire deux serveurs pour réaliser le partage des résultats de préservation et d'évaluation de la confidentialité entre les différents demandeurs. Il s'agit d'un scénario avec deux parties de service indépendantes qui ne se croisent pas. L'un d'eux est un mandataire autorisé (AP) qui est responsable du contrôle de l'accès et de la gestion des éléments de preuve agrégés afin d'améliorer la vie privée des entités évaluées. L'autre est une partie à l'évaluation (PE) (p. ex., offerte par un fournisseur de services Cloud) qui traite les données recueillies auprès d'un certain nombre de fournisseurs de preuves de confiance. Le PE traite les données collectées sous une forme chiffrée et produit un résultat de pré-évaluation de confiance chiffré. Lorsqu'un utilisateur demande le résultat de pré-évaluation à partir du PE, le PE vérifie d'abord l'admissibilité de l'utilisateur avec AP. Si le contrôle est positif, l'AP rechiffre le résultat de pré-évaluation qui peut être déchiffré par le demandeur (schéma 1) ou il y a une étape supplémentaire impliquant le PE qui empêche l'AP d'obtenir le résultat clair de pré-évaluation tout en permettant le décryptage du résultat de pré-évaluation par le demandeur (schéma 2) [74].

Améliorer le modèle de protection de la vie privée du K-anonymat et de la L-diversity

— **Anonymisation basée sur MapReduce** : pour un traitement efficace des données, un framework MapReduce est proposé. Les données sont divisées en segments de taille égale qui sont ensuite insérés dans un mappeur distinct. Les mappeurs traitent ses morceaux et fournissent des paires comme sorties. Les paires ayant la même clé sont transférées par le framework à un réducteur. Les ensembles de sortie du réducteur sont ensuite utilisés pour produire le résultat final [32, 34].

— **K-anonymat avec MapReduce** : étant donné que les données sont automatiquement divisées dans le cadre de MapReduce, l'algorithme de k-anonymat doit être insensible à la distribution des données entre les mappeurs. Pour réduire le nombre

des itérations requises, chaque classe d'équivalence est divisée en (au maximum) classes d'équivalence q dans chaque itération, plutôt que seulement deux [71].

— **MapReduce basé sur la L-diversité** : L'extension du modèle de protection de la vie privée k-anonymity à l-Diversity nécessite l'intégration des valeurs sensibles dans les clés de sortie. Par conséquent, les paires générées par les mappeurs et les combineurs doivent être modifiées de manière appropriée. Contrairement au mappeur dans k-anonymity, le mappeur dans l-diversity, reçoit à la fois des quasi-identificateurs et d'attributs sensibles comme entrées [71].

Anonymisation rapide des gros flux des données :

Le Big Data associé à un timestamp est appelé le Big Data stream. Les données du capteur, les enregistrements du centre d'appels, les flux de clics et les données de santé sont des exemples de flux de données massives. Les paramètres de qualité de service (QoS) tels que le délai d'acheminement des paquets de bout en bout, la précision et le traitement en temps réel ont quelques contraintes du traitement des flux de données massives. [75].

L'une des approches courantes pour anonymiser les données statiques est le k-anonymat. Cette approche n'est pas directement applicable pour l'anonymisation des flux Big Data. Les motifs sont les suivants [76] :

1. Contrairement aux données statiques, les flux de données Big Data ont besoin d'un traitement en temps réel et les approches existantes de k-anonymat sont NP-Hard, comme illustré.
2. Pour que les algorithmes statiques existants de k-anonymat réduisent la perte d'information, les données doivent être numérisées à plusieurs reprises pendant la procédure d'anonymisation. Le même processus est impossible dans le traitement des flux de données.
3. Les flux de données qui doivent être anonymisés dans certaines applications augmentent énormément.

Les flux de données Big Data sont devenus si importants que leur anonymisation devient un défi pour les algorithmes d'anonymisation existants. Pour faire face aux premier et deuxième défis susmentionnés, l'algorithme "Flocking anomalies in data streams" FADS [77]

a été choisi. Cet algorithme est le meilleur choix pour l'anonymisation des flux de données. Mais il a deux inconvénients majeurs :

1. L'algorithme FADS gère les tuples séquentiellement donc n'est pas adapté pour le flux de Big Data.
2. Certains tuples peuvent rester dans le système pendant un certain temps et sont déchargés quand un seuil spécifié arrive à sa fin.

Ce travail a apporté trois contributions. Tout d'abord, en utilisant le parallélisme pour étendre l'efficacité de l'algorithme FADS et le rendre applicable à l'anonymisation du flux de Big Data. Deuxièmement, la proposition d'une heuristique proactive simple pour empêcher la publication d'un tuple après son expiration. Troisièmement, illustrer (par des résultats expérimentaux) que FAST est plus efficace que FADS et d'autres algorithmes

existants, tout en réduisant de manière notable la perte d'informations et la mesure des coûts au cours du processus d'anonymat.

Heuristique proactive : Dans le système FADS, un nouveau paramètre est considéré comme représentatif du délai maximal tolérable pour une application. Ce paramètre s'appelle "expiration-time". Pour éviter qu'un tuple ne soit publié lorsque son délai d'expiration est écoulé, une méthode heuristique simple est définie. Dans FADS, il n'y a pas de vérification pour savoir si un tuple peut rester plus dans le système ou non. Par conséquent, certains tuples sont publiés après expiration. Ce problème viole la condition temps réel d'une application de flux de données et augmente également le coût de la mesure.

La protection de la vie privée Big Data : exemple des données de santé

La nouvelle vague de numérisation des dossiers médicaux a connu un changement de paradigme dans l'industrie de la santé. Par conséquent, l'industrie des soins de santé est témoin d'une augmentation du volume de données en termes de complexité, de diversité et de rapidité [78, 79]. Le terme «Big Data» fait référence à des ensembles de données importantes et complexes, qui dépassent les capacités de calcul, de stockage et de communication existantes des systèmes conventionnels. Dans le domaine de la santé, plusieurs facteurs donnent l'impulsion nécessaire pour exploiter la puissance du Big Data [80].

L'exploitation de la puissance d'analyse Big Data et de la recherche avec un accès en temps réel aux dossiers des patients pourrait permettre aux médecins de prendre des décisions éclairées sur les traitements [81]. Le Big Data obligera les assureurs à réévaluer leurs modèles prédictifs. La surveillance à distance en temps réel des signes vitaux à l'aide de capteurs intégrés (attachés aux patients) permet aux fournisseurs de soins de santé d'être alertés en cas d'anomalie. La numérisation des soins de santé par l'analyse intégrée est l'une des prochaines grandes vagues dans le domaine des technologies de l'information dans le domaine des soins de santé (IT), les dossiers de santé électroniques (DSE) étant un élément essentiel de cette vision. Avec l'introduction de programmes d'encouragement à la HER [82], les organismes de soins de santé ont reconnu la proposition de valeur du DSE visant à faciliter un meilleur accès à des données complètes, précises sur les soins de santé. Avec l'environnement de risque, on constante l'évolution et l'introduction de nouvelles menaces et vulnérabilités émergentes, les violations de la sécurité devraient augmenter au cours des prochaines années [83].

Le Big Data a permis de faire une enquête exhaustive sur les différents outils et techniques utilisés dans les soins de santé omniprésents d'une manière spécifique à la maladie. Cela couvre les principales maladies et troubles qui peuvent être rapidement détectés et traités avec l'utilisation de la technologie, telles que les chutes (mortelles et non mortelles), la maladie de Parkinson, les troubles cardio-vasculaires, le stress, etc.

L'adoption du Big Data dans le domaine de la santé augmente considérablement les préoccupations en matière de la sécurité et la protection de la vie privée. Au début, les informations sur les patients sont stockées dans des centres de données avec différents niveaux de sécurité. Les solutions de sécurité traditionnelles ne peuvent pas être appliquées directement à des ensembles de données importants et intrinsèquement diversifiés. Avec l'augmentation de la popularité des solutions cloud de soins de santé, la complexité de la sécurisation massive des solutions logicielles distribuées en tant que service (SaaS) augmente avec différentes sources et formats de données. Par conséquent, la gouvernance du Big Data est nécessaire avant d'exposer les données à l'analytique.

Gouvernance des données

1. Alors que l'industrie de la santé s'oriente vers un modèle d'affaires fondé sur la valeur en tirant parti de l'analyse des soins de santé, la gouvernance des données sera la première étape de la réglementation et de la gestion des données sur les soins de santé.
2. L'objectif est d'avoir une représentation commune des données qui englobe les standards de l'industrie et les normes locales et régionales.
3. Les données générées par "Boundary Sensitive Network" BSN [84] sont de nature diverse et nécessiteraient une normalisation, une normalisation et une gouvernance avant l'analyse.

L'analyse des risques pour la sécurité et la prévision des sources de menaces en temps réel sont de la plus haute nécessité dans l'industrie florissante des soins de santé. En effet, l'industrie de la santé est témoin d'un déluge d'attaques sophistiquées allant du déni de service distribué (DDoS) aux logiciels malveillants furtifs. L'industrie de la santé mise sur les technologies émergentes du Big Data pour prendre des décisions mieux informées. L'analyse de la sécurité est au cœur de toute conception de la solution SaaS basée sur le Cloud hébergeant des informations de santé protégées [85]. L'invasion de la confidentialité des patients est une préoccupation croissante dans le domaine de l'analyse des données massives. Les schémas de chiffrement qui préservent la confidentialité et qui permettent d'exécuter des algorithmes de prédiction sur des données chiffrées tout en protégeant l'identité d'un patient sont essentiels à l'analyse des soins de santé [86].

2.5 Conclusion

Dans ce chapitre, nous avons étudié les défis posés par la préservation de la vie privée pour les données massives en identifiant d'abord les exigences privacy des données volumineuses, puis en discutant si les techniques existantes de préservation de la vie privée sont suffisantes pour le traitement des données volumineuses. Les défis posés [43] sont présentés, ainsi que les avantages et les inconvénients des technologies existantes de préservation de la vie privée dans le contexte des applications de données volumineuses.

Chapitre 3

Protocole pour la préservation de la confidentialité dans le système distribué

3.1 Introduction

Les Big Data posent d'énormes défis et des sujets de recherche, l'un des principaux sujets de recherche est comment préserver la confidentialité [87,88] sur ces données. L'idée de protéger la confidentialité des données volumineuses stockées sur différents sites ou sur des systèmes distribués est très importante, car elle engendrera des graves conséquences si les données sensibles sont divulguées.

Cela peut impacter l'image de marque de l'entreprise et la perte des données sensibles. Dans ce chapitre, nous proposons une solution pour le traitement des données sensibles de manière anonyme.

L'objectif principal de cette contribution est de répondre à un enjeu majeur : la confidentialité tout en tenant compte de la protection de la vie privée. L'objectif de notre travail est d'assurer la confidentialité du traitement des données du groupe des agences bancaires qui doivent centraliser le traitement des données sensibles et confidentielles de leurs utilisateurs qui doivent rester anonymes. Nous explorons un nouveau système de traitement de données anonyme et sécurisé. Nous proposons que chaque agence bancaire utilise un pseudonyme vérifiable et irréalisable [89,90] et nous éliminons le système d'authentification centralisé. Avec cette technique, nous préservons l'anonymat des agences et de leurs clients, puis nous utilisons le chiffrement des données pour les rendre confidentielles. Le reste du présent document est organisé comme suie : la deuxième section fournit les exigences et les options de notre solution, le protocole proposé est donné à la troisième section. Enfin, nous terminons ce chapitre avec des présentations de quelques résultats expérimentaux dans la section quatre.

3.2 Les exigences du protocole PPDS

Dans ce chapitre, nous nous concentrons sur les principales propriétés de la confidentialité et de la protection de la vie privée dans le traitement des données. Dans notre contexte, nous avons besoin de confiance, de la confidentialité et de la protection de la

vie privée qui peuvent être incompatibles avec partage et traitement en masse des données.

Dans ce chapitre nous revenons sur la définition de la confidentialité en tant qu'objectif de sécurité. Elle permet de renseigner sur la capacité d'utiliser une ressource ou un service sans avoir à révéler l'identité de l'utilisateur. Dans Big Data, certaines techniques de confidentialité peuvent être utilisées comme le chiffrement.

Afin d'assurer un traitement anonyme et réparti des données bancaires, nous pouvons remplacer les identités réelles des agences bancaires par un pseudonyme unique et non falsifiable. Dans notre cas, il devrait être possible de préserver la confidentialité des agences bancaires en utilisant un système d'authentification anonyme et sécurisé. Les données ne doivent être accessibles qu'aux entités autorisées. Dans ce chapitre, nous nous concentrons sur le traitement et le croisement des données anonymes des agences bancaires.

Notre traitement des données doit rester fiable sans tierce partie. Une confiance élevée peut être obtenue en utilisant un paradigme d'authentification et d'échange de clés de session. Pour ce faire, il faut gérer en toute sécurité (générer, stocker et utiliser) les informations d'identification des agences bancaires. Ce mécanisme doit etre flexibile et évolutif. Dans notre contexte de traitement de données distribuées va sans aucun doute à exposer l'évolutivité comme une exigence principale dans notre conception.

3.3 Protocole PPDS

Dans notre scénario, nous supposons que nous avons trois agences bancaires : UBS, Bank of America et OCBC. Nous supposons que les clients demandent un prêt d'une seule agence. Avant d'accorder le prêt, l'agence doit vérifier les actifs nets de ses clients, c'est-à-dire la somme de leurs liquidités dans d'autres banques. Le défi est de savoir comment obtenir toutes les informations nécessaires pour décider en conséquence si le prêt est possible ou non tout en respectant l'anonymat. Pour faire face à ce problème, nous proposons de combiner tous les attributs des clients dans un seul fichier. Pour le reste de ce chapitre, nous considérons le noeud comme l'agence bancaire et le client de la banque comme l'utilisateur.

Dans cette section, nous donnons un aperçu de notre protocole proposé et nous discutons de ses trois principaux concepts :

1. La confidentialité des noeuds et l'échange des clés.
2. Le choix du nœud en-tête .
3. Le croisement des données.

3.3.1 Présentation de la conception

Nous supposons que chaque client a un identifiant unique (CUI) tel que son numéro de passeport et ses attributs (Cash 1, Cash 2, Cash 3), comme indiqué dans le tableau ci-dessous Figure 3.1.

Les attributs sont collectés par les agences bancaires. Nous considérons que toutes les agences sont des semi-honnêtes et leurs puissance informatique élevée. Notre but dans

UID	Cash1	Cash2	Cash 3
xxx1A	V11	V12	V13
xxx2A	-	V22	V23
xxx3A	V31	V32	-
xxx4A	V41	V42	V43
xxx5A	V51	-	V53
...	...	...	...

FIGURE 3.1 – On peut avoir des clients avec des attributs sans valeur qui sont marqués par "-"

est de fusionner les attributs dans un seul fichier en utilisant des données traitées afin que la vie privée des clients puisse être respectée. En fait, aucune de ces banques ne sera en mesure de déterminer les sources des autres données après la fusion de toutes les données des clients. Chaque agence peut prendre des décisions pour les demandes de prêt en vérifiant tous les attributs de son client et ses actifs nets comme indiqué dans la figure 3.2.

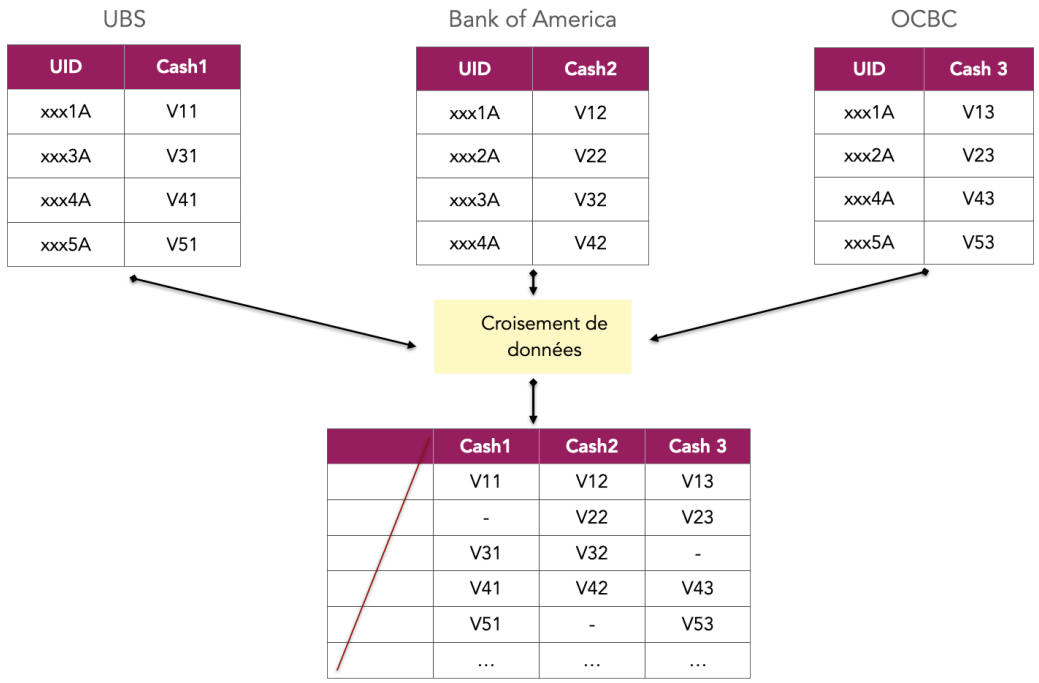


FIGURE 3.2 – La fusion des données recueillies à partir de différents nœuds.

La conception du protocole proposé s'articule autour de trois éléments clés : authentification des nœuds, sélection des nœuds et liaison de données. La première exigence dans cette conception est la confidentialité des données. Personne ne peut établir de lien entre les données et leurs sources. À cette fin, nous adaptons l'authentification basée sur la connaissance zéro dans un protocole peer-to-peer anonyme. Dans notre proposition, nous

devons sélectionner un nœud d'en-tête qui recevra et traitera les données. Pour cela, nous commençons par sélectionner le nœud principal qui sera le maître en charge de la collecte des données à traiter. Le nœud sélectionné recevra les données traitées des autres nœuds et les fusionner dans un fichier final.

3.3.2 Authentication

Dans cette section, nous présentons la partie authentification de notre contribution. Nous avons déjà étudié de nombreuses solutions d'authentification. La principale méthode d'authentification qu'on a retenue et la plus appropriée à notre contexte est le paradigme Pseudo Trust [90]. En fait, le protocole Pseudo Trust est utilisé pour l'authentification entre pairs dans le système P2P. Dans notre proposition, nous apportons quelques modifications à ce protocole en ajoutant la réputation des nœuds et les phases d'échange des fichiers qui ne sont pas conformes à notre contexte du système distribué.

Notre solution d'authentification est un paradigme de preuve à divulgation nulle de connaissance [91–93]. Chaque pair doit générer un pseudonyme vérifiable en utilisant l'identité réelle du nœud en utilisant une fonction de hachage unidirectionnel. Chaque nœud peut créer une authentification avec tous les autres nœuds sans divulgation d'informations sensibles. Pour rejoindre le groupe et communiquer avec d'autres parties, chaque nœud doit générer deux valeurs : une identité pseudo (P I) et un pseudo-certificat d'identité (P I C). Le (P I) permettant à son propriétaire de s'identifier dans le groupe de nœuds sans révéler sa véritable identité et il peut communiquer avec les autres nœuds.

La simple génération de la valeur PI peut entraîner une usurpation d'identité malveillante dans le système. Pour assurer une meilleure protection, chaque nœud doit générer sa valeur (PIC). Le nœud qui possède la combinaison appropriée (PI, PIC) est accepté dans le groupe.

Génération de PI et de PIC

1. Pour générer (P I) et (P I C) : la première étape consiste à choisir deux nombres premiers importants $\underset{1}{p}$ et $\underset{2}{p}$ et à calculer l'entier $n = \underset{1}{p} \cdot \underset{2}{p}$ puis à appliquer un prototype de fonction de hachage, puis à calculer $\underset{A}{Seed}$ par :

$$\underset{A}{Seed} = \underset{1}{h}(ID, \underset{1}{p}, \underset{2}{p})$$

   Tels que : $\underset{1}{h} : \{0,1\}^* \times \underset{n}{\mathbb{Z}^*} \times \underset{n}{\mathbb{Z}^*} \rightarrow \{0,1\}^m$

2. Après le calcul de $\underset{A}{Seed}$, le noeud A suit les étapes suivantes :
   (a) Tout d'abord, le noeud génère k entiers $j_1 \cdots j_k$ et calcule pour $i = 1$ à k, $v_{j_i} = h_2(Seed_A, j_i, n)(\mod n)$ avec j_i dans $J = j_1, ..j_k$
   (b) Calculer la plus petite racine carrée de $v_{j_i}(\mod n)$ noté v_{j_i} pour i entre 1 et k.
   (c) Calculer la valeur d'identité Pseudo :

$$PI_A = h_3(Seed_A, n)$$

   tels que $h_3 : \{0,1\}^m \times \mathbb{N} \rightarrow \{0,1\}^m$

3. Après avoir généré PI, le nœud génère son certificat :

$$PIC_A = \{PI_A, n, J, Seed_A\}$$

et le partager avec d'autres nœuds en le publiant dans un site public connu par tous les noeuds.

Initialisation des nouveaux nœuds

Après la génération de la pseudo-identité et du certificat, le nœud S tente de se connecter de façon anonyme à d'autres nœuds à l'aide de sessions anonymes.

Comme illustré sur la figure 3.3, pour assurer la connexion du noeud S au noeud R, un noeud de queue TS sera le relais de S vers le système de multi-diffusion. A la fin de ce processus d'initialisation, un autre noeud de queue TR reçoit le message.

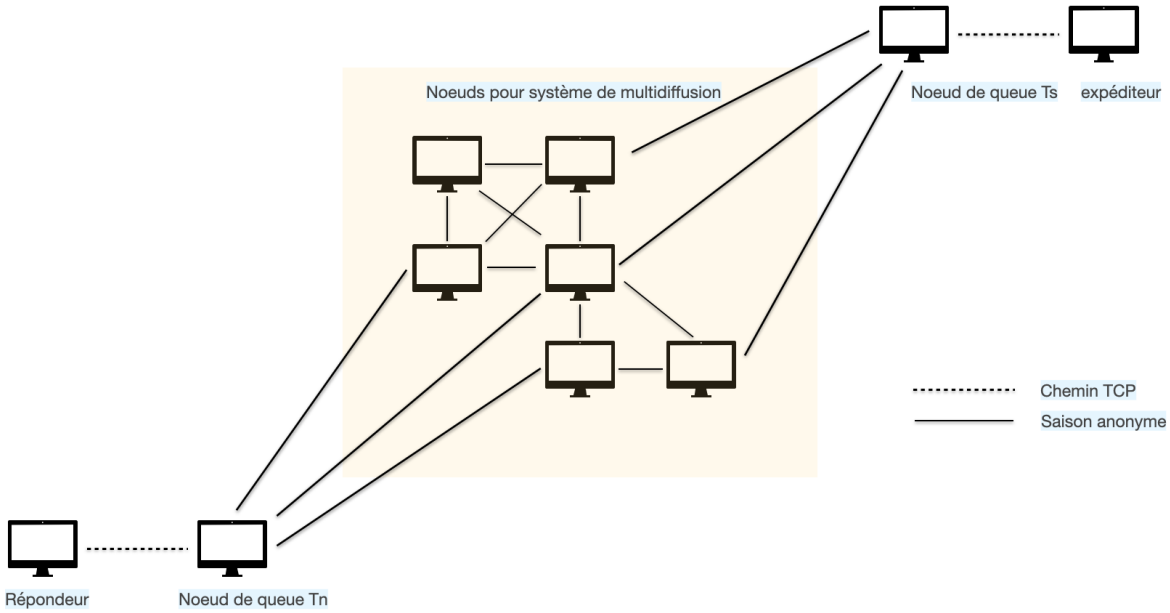


FIGURE 3.3 – Connexion entre deux nœuds

Initialisation des nouveaux nœuds

Après l'établissement de la session anonyme, nous indiquons S comme le noeud expéditeur qui diffuse la requête q via un chemin anonyme [94]. Lorsque le noeud récepteur indiqué R reçoit la requête, il répond par son noeud queue TR.

Trois paramètres publics g,P et Q sont publiés dans un serveur bootstrap. P et Q sont de gros nombres premiers de sorte que Q divise P - 1, g satisfait à cette équation gQ = 1modP et il est choisi parmi (1, P-1), pour assurer l'intégrité et la confidentialité des données échangées entre noeuds. Nous utilisons le protocole Diffie-Hellman [95] pour extraire des données de ce serveur.

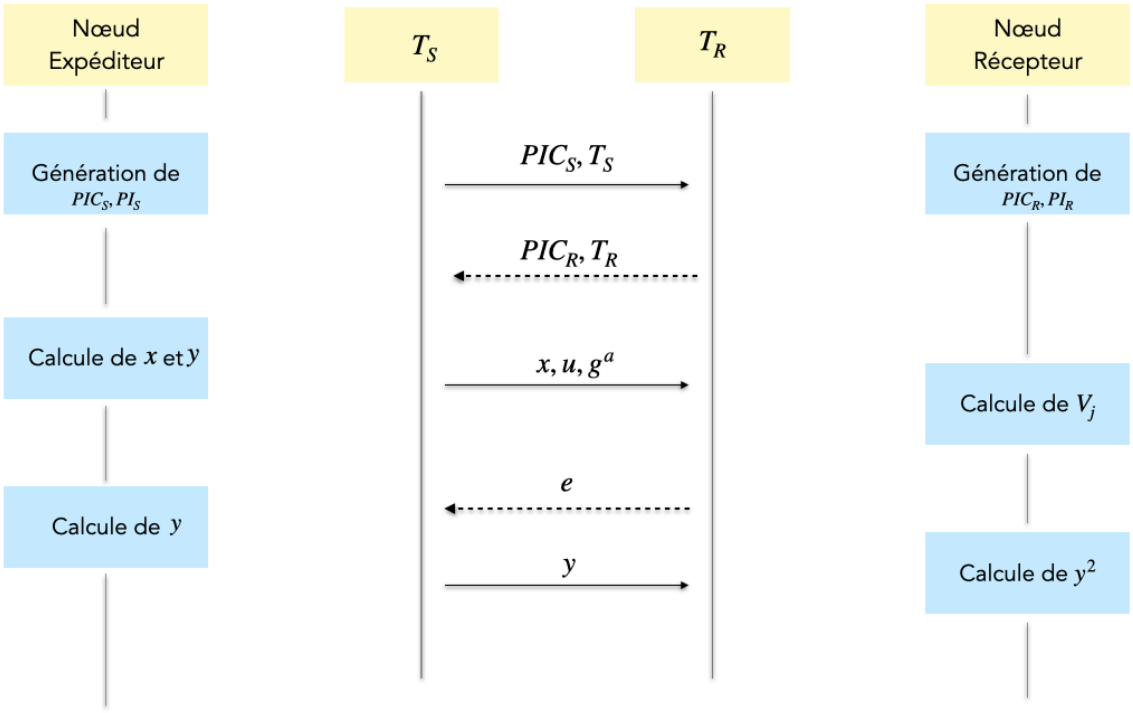


FIGURE 3.4 – Procédure d'authentification

Avant de commencer l'authentification, le noeud S envoie des requêtes contenant PICS, TS par son noeud terminal TS. Quand R reçoit la requête, il la relance avec une requête contenant son (PICR, TR) via TR comme illustré dans la figure 3.4.

L'authentification commence par l'envoi d'une demande d'authentification à R, puis la reception des réponses R avec une requête de validation de l'authentification, après cela, une preuve de connaissance nulle sera échangée entre S et R. Si le noeud de l'expéditeur prouve le défi, une nouvelle connexion sera établie entre les deux noeuds.

Pour plus de détails sur les messages échangés au cours de la phase d'authentification (voir la figure 3.4) :

1. Demande d'authentification : le nœud expéditeur génère trois nombres aléatoires : a, c et x tels que $a \in [1, Q)$,$c \in [0, n_S)$ et $x = c^2 mod n_S$ ensuite calculer $g^a \mod P$ et $u = h_4(x, PI_R, T_R, g^a)$ tels que $h_4 : \mathbb{Z}_n^* \{0,1\}^m \times \{0,1\}^* \times \mathbb{Z}^{*_p} \to \{0,1\}^k$ envoyer ensuite x, u, g^a à R

2. Vérification de la requête : le noeud R calcule $u' = h_4(x, PI_R, T_R, g^a)$ ensuite vérifie si $u = u'$. Si la vérification est confirmée, R passe à l'étape suivante, sinon la demande d'authentification sera rejetée.

3. Challenge e : Si R possède PI_S puis calculer $\{v_{j_i}\}^k$ de S tels que $v_{j,i} = h_2(Seed_S, j_j, n_S), j \in J_I$, $i = 1 \cdots k$ et génère un vecteur binaire aléatoire $e_j = (e_{j_1} \cdots e_{j_k}) \in \{0,1\}^k$ et l'envoie à S. Sinon R arrête la procédure d'authentification.

4. Génération de preuves : peer S répond avec y, tels que

$$y = c\left(\prod_{i=1}^{k} s_{j_i} e_{j_i} + u_i(\mod n_S)\right), \left\{S^{j^i}\right\}_S^k, i = 1 \cdots k$$

5. Vérification : le noeud R vérifie :

$$y^2 = x\left(\prod_{i=1}^{k} v_{j_i} e_{j_i} + u_i(\mod n_S)\right), \left\{v^{j^i}\right\}_S^k, i = 1 \cdots k$$

Pour l'échange de clés de session, lorsque le nœud S exécute la première étape (demande d'authentification), il choisit un nombre aléatoire $a \in [1, Q)$ et le garde secret, de l'autre côté, le nœud R choisit au hasard un numéro $b \in [1, Q)$ et garde cette valeur secrète. Lorsque l'authentification est terminée avec succès, le nœud S calcule $k = (g^b)^a \mod P$ et le nœud R calcule $k' = (g^a)^b \mod P$.

3.3.3 Sélection tête de nœud

Après l'authentification et les échanges de clés, chaque noeud parmi les n noeuds de telle sorte que $n > 2$ possède un (PIC) associé à (PI) et une clé symétrique avec chaque noeud existant.

Dans cette étape, chaque nœud doit générer une clé publique PbK et une clé privée PrK. Ensuite, le nœud ayant la plus petite PI est sélectionné par d'autres nœuds comme tête de nœud (NH). Enfin, la clé publique de NH dénotée par PbK_{NH} est diffusée aux autres nœuds et elle est utilisée en outre dans les opérations de chiffrement et de déchiffrement qui seront expliquées en détail dans la section suivante.

La tête de nœud préserve la confidentialité des données. Il reçoit les données chiffrées avec sa clé publique envoyées par tous les autres nœuds pendant le traitement des données.

3.3.4 Croisement des données

Après avoir fixé le nœud d'en-tête N H, chaque nœud i tel que i n ajoute une valeur appelée "**salt**" aux données. Cette méthode consiste à renforcer la sécurité de l'information par ajoutant des données supplémentaires, ce qui empêche qu'une donnée chiffrée provienne de deux Données identiques. Ensuite, tous les nœuds chiffrent leurs données avec la clé publique de N H. Pour chaque donnée $V_{i,l}$ dans le tableau illustré à la figure 3.5, il y a un nouveau salt généré aléatoirement, ensuite, tous les nœuds chiffrent leurs données avec la clé publique de NH Pbk_{NH} comme indiqué ci-dessous :

$$En(V_{i,l} + salt)_{NH}$$

Sachant que la valeur de "**salt**" ne s'applique pas sur la colonne UID.

Ensuite, un noeud i où $i \in [1 \cdots n] \setminus \{NH\}$ doit choisir un autre noeud j tel que $j \in [1 \cdots n] \setminus \{NH, i\}$ au hasard. Comme l'illustre la figure 3.6, au point (1), tous les noeuds Nh, A, B et C chiffrent leurs données $\{d_i\}$ dans la table avec Pbk_{NH} :

$$f_i = \{d_i\}_{Pbk_{NH}}$$

Noeud i	
UID	attribut
xxx1A	$V_{i,1}$
xxx2A	$V_{i,2}$
.	.
.	.
xxxlA	$V_{i,l}$
xxxnA	$V_{i,n}$

$\{d_i\}$

FIGURE 3.5 – Table de nœud i

Ensuite, le noeud B choisit au hasard le noeud C comme indiqué au point (2) dans la figure et chiffre le fichier de base de données f_B avec un algorithme de chiffrement symétrique AES avec la longueur de la clé 256 bits pour obtenir f'_B :

$$f'_B = Enc(f_B)_{k_{b,c}}$$

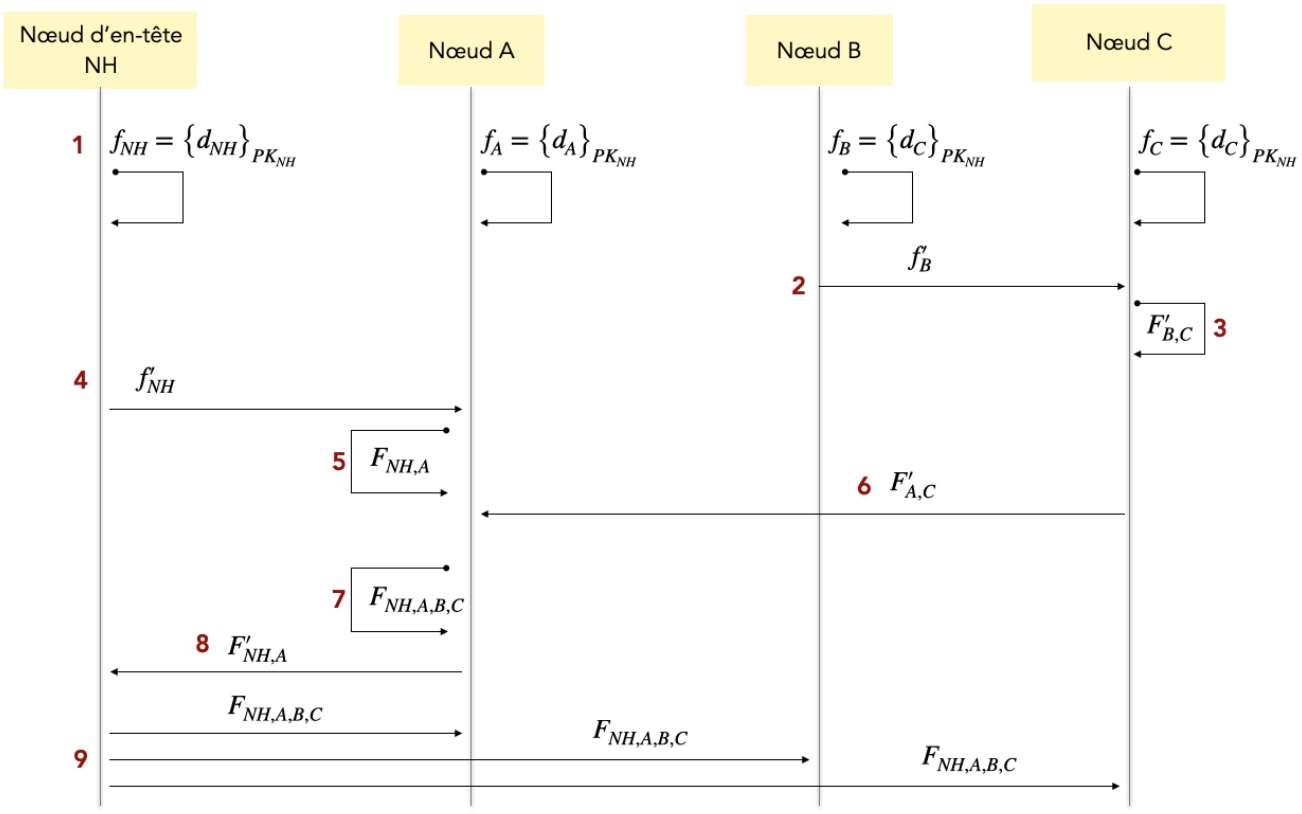


FIGURE 3.6 – Processus de croisement des données

L'algorithme AES-256 est utilisé par tous les noeuds avec les clés symétriques $k_{i,j}$ générées entre les noeuds dans l'étape d'authentification pour assurer la confidentialité lors de la transmission de fichiers.

Au point 3) : Le noeud C accepte f'_B et le décrypte avec le $k_{b,c}$ pour obtenir $f_B = \{d_B\}_{Pbk_{NH}}$ puis il applique le croisement des données avec la requête SQL et permute les colonnes :

$$F_{B,C} = \left(Permutation\left(Cross\left(\{d_B\}_{Pbk_{NH}}, \{d_C\}_{Pbk_{NH}}\right)\right)\right)$$

Simultanément, le noeud A reçoit un fichier chiffré $f'_{NH} = Enc\,(f_{NH})_{k_{NH,A}}$, A avec la clé symétrique du noeud NH comme au point (4) et exécute le même processus que B et C, le noeud A contient $F_{NH,A}$ point (5) :

$$F_{NH,A} = \left(Permutation\left(Cross\left(\{d_{NH}\}_{Pbk_{NH}}, \{d_A\}_{Pbk_{NH}}\right)\right)\right)$$

Au point 6) : Le noeud A reçoit $F_{B,C}$ de C chiffré avec $k_{a,c}$:

$$F'_{a,c} = Enc(f_{B,C})_{k_{a,c}}$$

Après cela, il décrypte et applique le croisement des données et la permutation des colonnes, puis il supprime la colonne UID utilisée pour les données de croisement pour obtenir $F_{nh,a,b,c}$ voir point (7) :

$$F_{nh,a,b,c} = Permutation\,(Cross\,(F_{NH,A}, F_{B,C}))$$

et chiffré par $k_{a,nh}$ voir point (8) :

$$F'_{nh,a} = Enc\,(F_{nh,a,b,c})_{k_{a,nh}}$$

Enfin, le NH reçoit le fichier $F'_{nh,a}$ puis il le dechiffre pour obtenir $F_{nh,a,b,c}$ ce dernier noeud utilise sa clé privée Prk_{NH}.

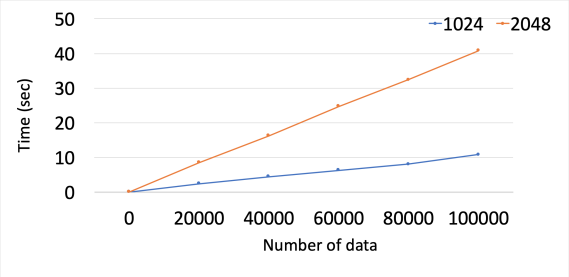
3.4 La mise en œuvre et la validation

Dans cette section, nous présentons la mise en oeuvre du protocole PPDS. Nous utilisons quatre tables contenant des données massives pour des expériences. Toutes les expériences ont été réalisées sur quatre serveurs amazons répartis sur différents sites (Ohio, Pekin, Mumbai et Singapour), avec les caractéristiques suivantes : Intel Core i7 et 32 Go de RAM. Les serveurs se sont interconnectés avec un réseau à grande surface (WAN). Chaque nœud génère une valeur PI. Supposons que l'ID du nœud Bank of America est $BankofAmericaNewYork2016$ et son PI unique est :

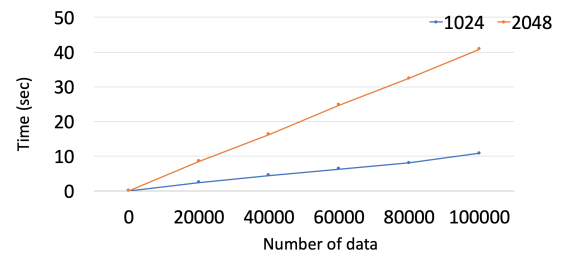
$$6675D73E3EBCDD52CDD9A2BFB0F064B082260BE1$$

L'algorithme de chiffrement asymétrique utilisé est RSA pour chiffrer les données dans chaque table, la taille des modules RSA est souvent controversée. Dans une utilisation commune, l'utilisation de modules de 1024 bits est généralement considérée comme suffisante pour assurer une sécurité pratique. Selon les recommandations du NIST [96] la taille minimale du module est de 2048 bits, pour une utilisation avant 2030 .

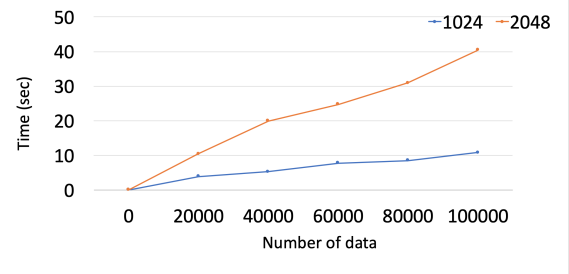
L'algorithme de chiffrement symétrique utilisé est AES-256 pour assurer la confidentialité des échanges de fichiers. La valeur "**salt**" est générée pour chaque donnée avec 16 octets de longueur et elle est limitée par les symboles /@ et @/.



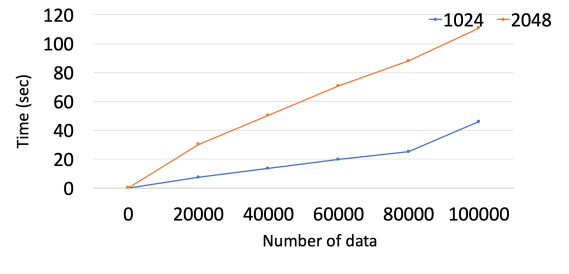
(a) Nœud 1 situé dans l'Ohio



(b) Nœud 2 situé à Pekin



(c) Nœud 3 situé à Mumbai



(d) Nœud 4 situé à Singapour

FIGURE 3.7 – Analyse de protocole

Les figures 3.7a, 3.7b et 3.7c montrent le temps requis et l'évolutivité pour quatre nœuds pour chiffrer ses données avec une clé publique. Nous avons utilisé deux clés publiques ayant comme longueur (1024 et 2048) pour le nœud d'en-tête.

Enfin, le nœud d'en-tête dechiifrement les données et les envoie à d'autres nœuds. La figure 3.7d illustre le temps nécessaire pour assurer cette opération.

Le temps est un facteur très important dans ce protocole, nous essayons maintenant d'optimiser le temps de calcul en utilisant la technique de multitraitement pour être mis en œuvre efficacement sur l'API Hadoop.

3.4.1 Comparer PPDS avec ses exigences

Dans cette section, nous évaluons les exigences et les options du PPDS qui ont déjà été cité au début du chapitre.

1. Confidentialité : Par conception, le protocole PPDS assure la confidentialité des nœuds dans la phase d'authentification. En fait, chaque nœud est authentifié par son PIC et son PI. Il n'y a pas de lien entre ces informations d'identification et l'identité du nœud. En outre, la sélection des nœuds d'en-tête et les données croisées sont effectuées indépendamment de l'identité des nœuds. Les données traitées sont chiffrées. Le même niveau de confidentialité est maintenu dans les trois phases.
2. Fiable sans tiers de confiance : la confidentialité est obtenue aussi en utilisant le chiffrement des données. Par conception, il n'y a pas de tiers impliqué dans notre architecture, mais la confiance est maintenue et validée dans l'authentification.
3. Flexibilité et évolutivité : l'objectif de flexibilité et d'évolutivité est atteint en utilisant un environnement informatique distribué. L'utilisation des solutions Amazon

et Cloud est le moyen approprié pour atteindre cet objectif.

Nous avons déjà proposé une solution pour le traitement anonyme des données. Nous sommes concentrés sur deux problèmes principaux : comment fusionner les données stockées dans les différentes agences de la Banque avec la préservation de la confidentialité des clients. Ensuite, aucun tiers de confiance ne soit impliqué dans ce processus.

Nous proposons également une validation formelle du protocole PPDS et nous fournissons une analyse de sécurité utilisant l'analyse de l'attaque de l'homme dans le milieu et de l'entropie comme solution pour détecter l'attaque par déni de service. Une analyse des performances est également fournie.

3.5 Validation formelle

Dans cette section, nous simulons notre protocole pour la vérification formelle des proprietes de sécurite à l'aide de l'outil AVISPA [97].

Le protocole est spécifié dans un langage de haut niveau HLPSL. La spécification est réécrite dans un format IF intermédiaire à l'aide d'un convertisseur. AVISPA utilise quatre autres outils (back-ends) qui prennent comme entrée le format IF, et qui offre la possibilité de faire 4 analyses différentes du même protocole. Toute langue convertible IF peut être vérifiée par AVISPA.

Nous définissons quatre rôles d'agent : a pour l'en-tête de noeud sélectionné par d'autres noeuds : b, c et d. La figure 3.12 illustre une partie de notre spécification HLPSL pour modéliser les rôles des agents.

Les objectifs de sécurité dans le processus de liaison de données est d'assurer la confidentialité dans le traitement et dans l'échange de fichiers entre les noeuds.

Les propriétés de sécurité suivantes ont été validé à l'aide d'AVISPA :

1. La confidentialité du fichier de données f_D est validé à l'aide `HLPSL`
2. Lors de l'accomplissement de la procédure de croisement et de la permutation dans le noeud `D`, le secret du fichier de données $F_{C,D}$ est validé à l'aide du fait `secret(Fcd', sec_3, {})`
3. Le nœud `A` chiffre leurs données et les envoie au nœud `B`, le secret du fichier de données f_A est validé à l'aide `secrecy_of` et le fait correspondant `secret(Fa', sec_1, {})`
4. Le secret final du fichier de données $F_{A,B,C,D}$ est validé en utilisant `secret(Fabcd', sec_4, {})`

La figure 3.8 valide l'intégralité de la sécurité des messages échangés définis dans le protocole.

3.6 Analyse de sécurité

3.6.1 L'attaque de l'homme au milieu

Cette section est consacrée à l'analyse de sécurité de notre protocole et un scénario d'attaque est discuté. Nous prouvons ici la résistance de notre proposition à l'attaque de

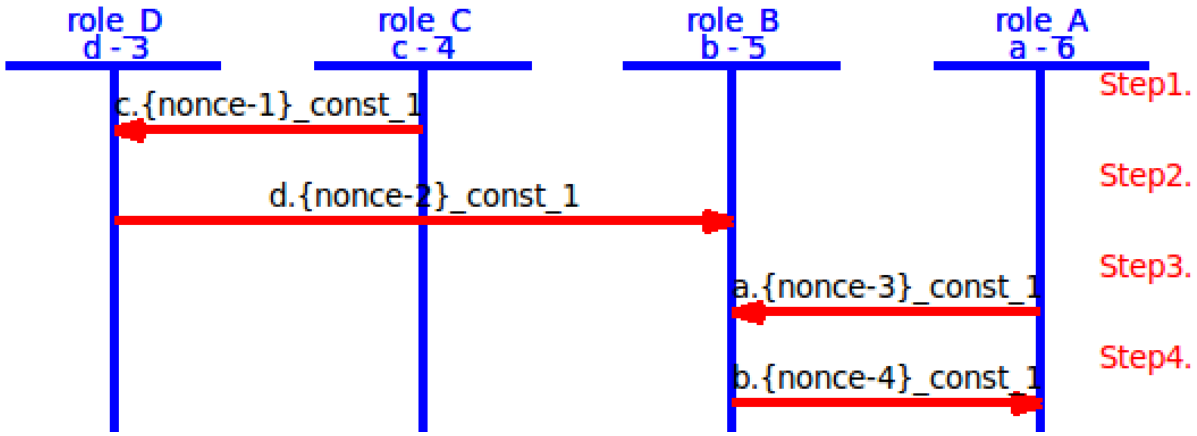


FIGURE 3.8 – Diagramme de séquence

l'homme du milieu (MITM) [98]. Cette attaque vise à intercepter les communications entre deux parties sans que l'une ou l'autre des parties sache que le lien entre elles a été compromis. Ici, nous nous concentrons sur la phase d'authentification du protocole. Comme le montre la figure 3.9, le noeud malveillant nommé M émet simultanément l'identité des deux victimes, il existe deux scénarios possibles pour cette attaque :

— Premier cas : le noeud du récepteur R ne reçoit pas la requête du noeud de l'expéditeur S.

— Deuxième cas : R reçoit la requête de S.

Dans le premier cas, M intercepte la requête q, la modifie avec q' et l'envoie à R. Si le noeud Malveillant modifie le PIC_S de l'expéditeur avec sa propre valeur PIC_M, alors R devient capable de savoir que le PIC_M ne correspond pas à la valeur PI_S.

Si le noeud malveillant modifie certains paramètres dans le contenu de la requête $q = (PIC_S, T_S)$ en $q' = (PIC_I, T_M)$, comme indiqué dans la figure 3.9, après avoir reçu q', le noeud R répond avec $r = (PIC_R, T_R)$, le noeud suspecté modifie la réexécution avec $r' = (PIC_R, T_M)$ et l'envoie à S, à la réception de r, ce dernier choisit r', puis c et $x = c^2 \bmod n$, $G = g^a \bmod P$ et $u = h4(x, PI, T, G)$ les envoie ensuite à R. M qui intercepte ces trois valeurs x,G,u et calcule $u' = h(x, PI_R, T_R, G)$, il s'appuie sur le noeud du récepteur, si u' est différent de u, alors le processus d'intrusion échoue, sinon M tente de calculer u' en modifiant u puis G et R pour obtenir $h(x, PI_R, T_R, G) = h(x', PI_R, T_R, G)$, ce cas est impossible à calculer parce que nous pouvons choisir une fonction de hachage résistante à l'attaque de collision.

S et R poursuivent le processus d'authentification jusqu'à l'étape de la génération de preuve. Le noeud de l'expéditeur commence par générer une preuve y et la transmet à R après avoir reçu le défi de R. Le noeud malveillant peut intercepter cette requête, mais il ne peut pas le prouver parce que le noeud malveillant ne peut pas connaître la valeur de e, donc l'attaque échoue.

Dans un second cas, R reçoit de nombreuses requêtes avec le même PI_S, le noeud récepteur est conscient d'être attaqué, il peut simplement choisir au hasard un noeud et redémarrer la procédure d'authentification.

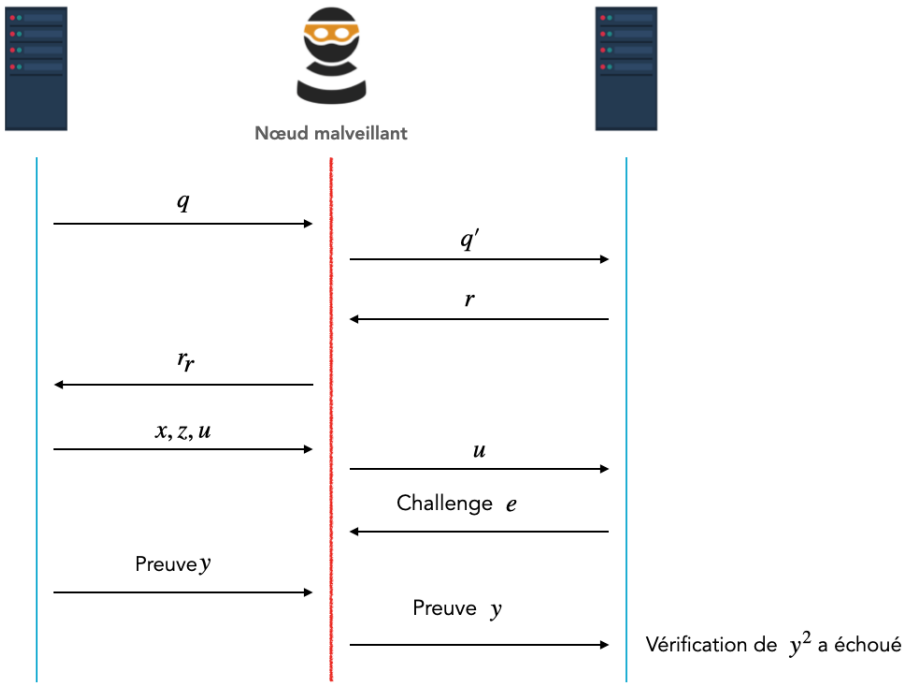


FIGURE 3.9 – Simulation d'attaque MITM

3.6.2 Entropie pour une attaque par déni de service

Pour distinguer entre attaque et trafic normal, les paquets échangés entre les agences peuvent être séparés en deux groupes qui sont : DATA et ACK. Ces paquets peuvent être communiqués par n'importe quelle agence de réseau. Pour donner une forme concrète de ce choix, nous définissons la variable aléatoire X où les paquets échangés définissent les valeurs possibles observées de cette variable.

$$\sum_{i=1}^{n} P(X = x_i) = 1 \tag{3.1}$$

En ce qui concerne la classification des paquets, nous pouvons écrire l'équation comme suit :

$$P(Data) + P(Ack) = 1 \tag{3.2}$$

Où $P(Data)$ et $P(Ack)$ présentent les probabilités respectives des paquets communiqués dont les types sont Data et Ack. En partant du principe que chaque paquet est dans l'un des types cités, chaque type de paquet peut être échangé entre toutes les agences du réseau. De cette façon, le type de paquets de données sort du noeud n_1,n_2...n_i et ainsi de suite. Pour simplifier la notation, un paquet est caractérisé par son noeud émetteur n_i et son type {Data, Ack}.

Par conséquent, le paquet émis à partir du noeud n_i et le type de données est noté par $Packet(n_i, Data)$. Nous notons la probabilité que le paquet reçu sort du noeud i des noeuds N et un paquet Data noté par : $Packet(n_i, Data)$ et la probabilité que le paquet reçu sort du noeud i et un paquet Ack est $Packet(n_i, ack)$. Nous avons toujours quel que soit le comportement des noeuds du réseau :

$$\sum_{i=1}^{n} P(n_i \setminus Data) = 1 \tag{3.3}$$

$$\sum_{i=1}^{n} P(n_i \setminus Ack) = 1 \tag{3.4}$$

et le cas de transmission des données reçues d'un nœud i est noté :

$$P(n_i \setminus Data) = \frac{1}{N} \tag{3.5}$$

le paquet Ack est noté :

$$P(n_i \setminus Ack) = \frac{1}{N} \tag{3.6}$$

Dans le réseau attaqué par DOS [99], le noeud malveillant indiqué n_M occupe le canal de transmission en émettant plus de paquets de données et il est logique d'avoir :

$$P(n_M \setminus Data) \gg \frac{1}{N}$$

Nous pouvons définir l'entropie des paquets pour les données et les comptes, comme indiqué ci-dessous :

$$H_{Data}(X) = -\sum_{i=1}^{N} P(n_i \setminus Data) \log_2(P(n_i \setminus Data)$$

$$H_{Ack}(X) = -\sum_{i=1}^{N} P(n_i \setminus Ack) \log_2(P(n_i \setminus Ack)$$

Pour calculer $P(n_i \setminus Data)$ ou $P(n_i \setminus Ack)$ nous pouvons utiliser la définition mathématique suivante de deux événements A et B

$$P(A \setminus B) = \frac{P(A \cap B)}{P(B)}$$

alors nous avons :

$$P(n_i \setminus Data) = \frac{P(n_i \cap Data)}{P(Data)} \tag{3.7}$$

$$P(n_i \setminus Ack) = \frac{P(n_i \cap Ack)}{P(Ack)} \tag{3.8}$$

Après avoir utilisé la formule donnée par l'équation (3.7) et (3.8), nous sommes capables de calculer l'entropie des paquets pour chaque type de paquet.

3.7 Évaluation de la performance

Pour évaluer la performance du protocole PPDS en cas d'attaque par déni de service, nous avons utilisé Shadow Simulator pour simuler notre topologie réseau et la communication entre les noeuds. Nous avons simulé deux scénarios différents avec 4 noeuds. Le premier scénario consiste à tester le comportement d'une communication réseau normale et le second à tester le comportement du trafic en cas d'attaque DoS. Pour chaque scénario, nous mesurons l'entropie des paquets Data et Ack. Pour un environnement plus réaliste, le simulateur Shadow a été configuré pour être utilisé dans un environnement de virtualisation de Amazon Web Service et pour des résultats plus concrets, nous avons répété l'expérience 10 fois pour approcher des résultats réalistes. Dans un trafic normal dans le réseau, tous les noeuds ont la même valeur de probabilité pour envoyer un paquet de données. En théorie, l'entropie de DATA H_{Data} a été calculé comme suit :

$$H_{Data} = -\sum_{i=1}^{4} \frac{1}{4} \log_2(\frac{1}{4}) = \log_2(4) = 2 \tag{3.9}$$

Comme l'illustre la figure 3.10, les valeurs de H_{Data} sont inférieures aux résultats théoriques, ce qui est correct parce que les valeurs théoriques de H_{Data} ont été déterminé dans le cas d'une transmission parfaite.

Dans le scénario d'attaque DoS, nous voyons qu'un ou deux noeuds malveillants peuvent consommer seuls deux tiers, et tous les autres noeuds sont honnêtes. Le H_{Data} peut être calculé comme suit :

$$H_{Data} = -\frac{2}{3} \log_2(\frac{2}{3}) - \sum_{i=1}^{3} \frac{1}{3*3} \log_2(\frac{1}{3*3}) = 1.44 \tag{3.10}$$

Dans notre expérience, la valeur de H_{Data} comme indiqué dans la figure 3.11 est toujours inférieure à la valeur calculée pour les mêmes raisons dans le trafic normal du réseau.

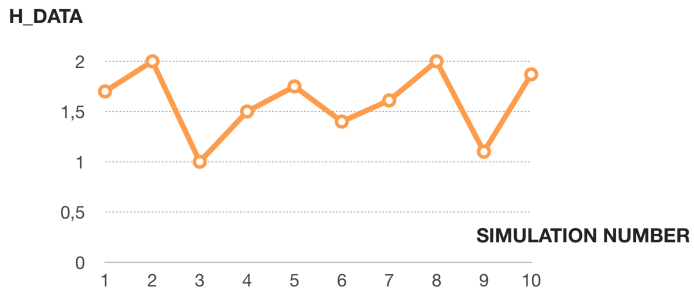


FIGURE 3.10 – Entropie des paquets de données dans un trafic normal

Le résultat de la simulation nous montre que H_{Data} est maximum dans le trafic normal et inférieur dans le réseau en cas d'attaque par déni de service. Nous pouvons conclure dans notre cas si H_{Data} est inférieur à 1, il y a un trafic suspect dans le réseau.

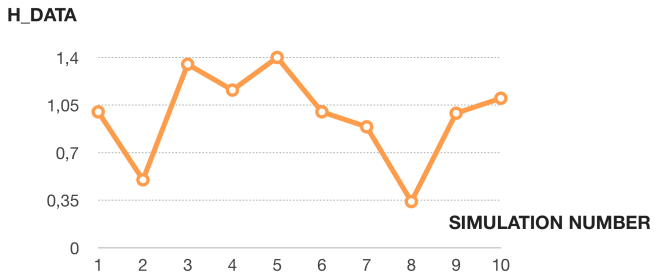


FIGURE 3.11 – Entropie des paquets de données dans le trafic en cas d'attaque par déni de service

3.8 Conclusion

Avec les progrès rapides du paradigme Big Data, la diversité de son utilisation et la multiplicité de ses applications, la sécurité devient un besoin complexe et urgent pour protéger tous les traitements de données. Dans ce chapitre, nous nous concentrons sur la confidentialité et le respect de la vie privée dans le traitement des données partagée dans le Big Data.

Nous présentons la mise en œuvre de la protection de la vie privée dans le protocole du système distribué (PPDS). Le protocole proposé a été formellement validé avec l'outil AVISPA. Nous avons utilisé l'analyse de sécurité pour l'attaque Man-In-The-Middle comme exemple et nous avons proposé une solution pour détecter les attaques DoS en utilisant le concept d'entropie. Le temps de réponse et le temps de traitement des données sont également analysés comme résultats de performance de notre preuve de concept. Dans ce chapitre, nous fournissons plus de détails et des résultats expérimentaux sur la mise en œuvre du protocole proposé. D'autres analyses des performances et des validations de sécurité seront également fournies.

```
role_A(A:agent,B:agent,C:agent,D:agent,Kab:symmetric_key,SND,RCV:channel
(dy))
played_by A
def=
     local
          State:nat,Fa:text,Fabcd:text
     init
          State := 0
     transition
          3. State=0 /\ RCV(start) =|> State':=1 /\ Fa':=new() /\
secret(Fa',sec_1,{}) /\ SND(A.{Fa'}_Kab)
          4. State=1 /\ RCV(B.{Fabcd'}_Kab) =|> State':=2 /\
secret(Fabcd',sec_4,{})
end role

role_B(A:agent,B:agent,C:agent,D:agent,Kab:symmetric_key,Kbd:symmetric_k
ey,SND,RCV:channel(dy))
played_by B
def=
     local
          State:nat,Fcd:text,Fa:text,Fabcd:text
     init
          State := 0
     transition
          2. State=0 /\ RCV(D.{Fcd'}_Kbd) =|> State':=1 /\
secret(Fcd',sec_3,{})
          3. State=1 /\ RCV(A.{Fa'}_Kab) =|> State':=2 /\
secret(Fa',sec_1,{}) /\ Fabcd':=new() /\ secret(Fabcd',sec_4,{}) /\
SND(B.{Fabcd'}_Kab)
end role

role_C(A:agent,B:agent,C:agent,D:agent,Kcd:symmetric_key,SND,RCV:channel
(dy))
played_by C
def=
     local
          State:nat,Fc:text
     init
          State := 0
     transition
          1. State=0 /\ RCV(start) =|> State':=1 /\ Fc':=new() /\
secret(Fc',sec_2,{}) /\ SND(C.{Fc'}_Kcd)
end role

role_D(A:agent,B:agent,C:agent,D:agent,Kcd:symmetric_key,Kbd:symmetric_k
ey,SND,RCV:channel(dy))
played_by D
def=
     local
          State:nat,Fc:text,Fcd:text
     init
          State := 0
     transition
          1. State=0 /\ RCV(C.{Fc'}_Kcd) =|> State':=1 /\
secret(Fc',sec_2,{}) /\ Fcd':=new() /\ secret(Fcd',sec_3,{}) /\ SND(D.
{Fcd'}_Kbd)
end role
```

FIGURE 3.12 – Rôles des agents dans HLPSL

Chapitre 4

Anonymisation des données : les traces NetFlow

4.1 Introduction

NetFlow est un protocole de surveillance de trafic réseau. Il a été initialement développé par Cisco et mis en oeuvre dans les routeurs des entreprises. NetFlow a commencé comme un cache pour améliorer les performances des recherches IP et il est ensuite devenu un outil de mesure de flux largement utilisé. Les routeurs exécutant NetFlow conservent un "cache de flux" incluant des enregistrements de flux qui collectent des détails sur le trafic transféré par le routeur. Ces enregistrements de flux peuvent être collectés, analysés et archivés par un ordinateur. Ce flux est exporté à l'aide du protocole User Datagram Protocol. Il s'agit de la principale technologie de construction de matrices de trafic de base. NetFlow est également utilisé comme système de détection des intrusions pour appliquer une politique de sécurité donnée, notamment pour identifier les attaques par déni de service. Malheureusement, cet analyseur de trafic est une arme à double tranchant. Souvent, dans les paquets capturés, l'en-tête révèle des informations sur l'identité de l'utilisateur, par exemple des e-mails chiffrés avec le sujet, l'expéditeur et le destinataire, en texte clair. Cette violation de la confidentialité peut être plus subtile en donnant des détails techniques sur l'environnement utilisé. Par exemple, un vidage complet des paquets peut fournir aux utilisateurs malveillants l'espace disque requis et les performances liées au réseau et à son équipement.

Ce chapitre fournit l'architecture sécurisée des données partagées en ce qui concerne les contraintes de confidentialité. Nous choisissons les journaux NetFlow comme étude de cas et proposons un nouveau service appelé "Privacy as a Service" qui vise à assurer la confidentialité dans le processus de flux réseau. Enfin, nous utilisons un modèle de confidentialité dynamique qui vise à répondre aux exigences des utilisateurs en matière de protection de la vie privée et aux défis de sécurité tout en partageant des données.

Afin de surveiller les violations de la sécurité, il est essentiel que le flux puisse remonter à la source d'une manière qui ne laisse aucun doute. Dans ce chapitre et dans la continuité du traitement de sujet "sécurité des données partagées", nous examinons deux objectifs : confiance et confidentialité des processus de traitement données. Une confiance élevée peut être obtenue en utilisant un élément fiable pour gérer en toute sécurité (générer, stocker

et utiliser) les informations d'identification des utilisateurs. La deuxième exigence est l'anonymat des utilisateurs. Notre recherche répondra à des questions telles que la façon dont les paramètres cryptographiques peuvent être utilisés pour sécuriser le partage des données en matière de confidentialité entre les utilisateurs anonymes et la manière efficace dont un utilisateur anonyme de sessions de communication anonymes peut communiquer sans compromettre l'anonymat des deux parties.

Des travaux antérieurs ont proposé des approches d'anonymisation et sont basés sur des techniques de randomisation ou de pseudonymisation. En effet, les données sont classées en fonction de leur niveau de confidentialité ; une évaluation qualitative dynamique des risques pour la confidentialité est utilisée. Par exemple, nous utilisons le K-anonymat dynamique pour le processus d'anonymisation. Le K-anonymat est réalisé à différentes étapes, comme la généralisation et la définition des quasi-identifiants. Nous utilisons ce concept pour ajouter les paramètres d'anonymisation. Dans cet article, nous explorons donc un nouveau schéma de traitement des données anonymes et sécurisées. En fonction des niveaux de confidentialité de chaque groupe des données, nous définissons le K pour K-anonymat et le schéma de pseudonymisation utilisé pour la généralisation.

L'organisation de ce chapitre est la suivante : nous commençons par un bref historique du travail connexe décrivant tous les schémas d'anonymisation. Dans la section trois, nous présentons le modèle de menace de ce travail. Ensuite, la solution proposée avec son paradigme d'anonymat connexe est fournie dans les sections quatre et cinq, suivies de quelques résultats expérimentaux. Dans la section six, nous utilisons NetFlow pour fournir une analyse de la protection des renseignements personnels etc.. . En fonction de nos constatations, nous discutons également des travaux futurs. Enfin, dans la section sept, nous résumons nos conclusions.

4.2 Contexte

Des travaux récents ont montré qu'il existe deux approches principales pour assurer l'anonymat des données. La première est la randomisation, qui modifie la véracité des données afin d'affaiblir le lien entre les données et l'identité de l'individu concerné et de rendre les données suffisamment incertaines pour qu'elles ne puissent plus être retracées par un individu particulier. Nous pouvons utiliser plusieurs méthodes telles que l'addition de bruit ou la permutation. Le second paradigme est l'anonymat, qui dilue les attributs des personnes impliquées en changeant leur échelle ou leur ordre de grandeur respectif (d'une échelle communautaire à une échelle régionale, par exemple). Il y a trois principaux paradigmes d'anonymisation : k-anonymat, L-diversité et confidentialité différentielle/ differential privacy. K-anonymat répond au risque de révélation d'identité ; elle vise à empêcher qu'une personne associée à une clé d'identification soit isolé en la regroupant, au moins, avec des individus. Pour ce faire, les attributs sont généralisés tel que tous les individus partagent la même clé d'identification. Généraliser signifie en fait "enlever un certain degré de précision" à certains domaines. Par exemple, l'abaissement de la granularité géographique d'une ville à une région inclut un plus grand nombre de personnes impliquées. La généralisation peut être globale (remplacement de toutes les villes par la région correspondante) ou locale (nous ne remplaçons que les petites villes par leurs régions, nous conservons les grandes villes où il y a au moins des lignes k - capitales, métropoles etc.). Plusieurs limitations ont été identifiées dans cette technique, principa-

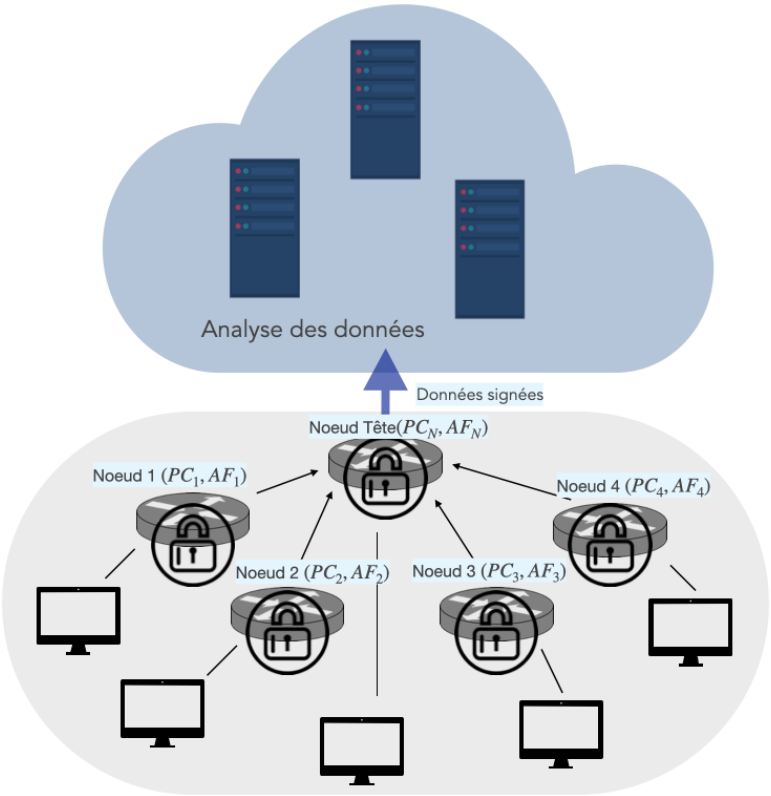


FIGURE 4.1 – L'architecture du privacy as a service

lement des attaques telles que la correspondance non triée, la libération complémentaire, la minimalité et les attaques temporelles [65, 100, 101].

L-diversity étend le K-anonymat pour s'assurer qu'il n'est plus possible d'obtenir certains résultats par des attaques par inférence, en s'assurant que dans chaque classe d'équivalence, chaque attribut a moins l valeurs différentes. Bien que cette technique offre une défense optimale contre les tentatives de révélation inférentielle lorsque les valeurs d'attribut sont correctement distribuées, il convient de souligner qu'elle n'empêche pas les fuites d'informations si les attributs d'un segment sont répartis dans le temps. La L-diversité peut être redondante et laborieuse à réaliser et elle est ouverte aux attaques de révélation d'attributs (risque de corrélation). Il est également sensible aux attaques de similarité et ne permet pas d'éviter l'exposition des attributs en raison de la relation sémantique entre les attributs sensibles [101, 102].

Enfin, dans le cas d'une differential privacy, les ensembles de données sont communiquées à des tiers en réponse à une demande spécifique plutôt que d'être publiés comme un ensemble de données unique. Les aperçus anonymisés sont produits au moyen d'un sous-ensemble de demandes adressées à un tiers. Le sous-ensemble comprend un bruit aléatoire délibérément ajouté a posteriori. Cette technique ne modifie pas les données d'origine. Par conséquent, le responsable du traitement des données demeure en mesure d'identifier les personnes dans les résultats des requêtes différentielles sur la protection de la vie privée. Malheureusement, il n'est pas possible d'atteindre un maximum d'intimité et un haut niveau de précision en même temps. Le differential privacy est obtenue par randomisation et donc des garanties maximales de protection de la vie privée qui ne sont

possibles qu'en ajoutant suffisamment de bruit pour réduire la précision.

4.3 Modèle de menace

Dans notre approche, pour assurer la sécurité et la protection de la vie privée, la solution proposée est basée sur un modèle "honnête mais curieux", ce qui signifie que toutes les entités de notre système (c.-à-d. les fournisseurs de services et les utilisateurs de services partageant des données) utiliseront notre système honnêtement, mais peuvent être curieux de recueillir et d'analyser certaines informations.

Nous ciblons deux attaques de K-anonymat. La première attaque est l'attaque d'homogénéité, qui exploite le cas où toutes les valeurs sensibles dans un jeu d'enregistrements K sont identiques. Dans tels cas, même si les données ont été K-anonymisées, la valeur sensible pour l'ensemble des K enregistrements peut être prédite.

La deuxième attaque est l'attaque de base de connaissances qui utilise une association entre un ou plusieurs attributs quasi-identificateurs et l'attribut sensible pour réduire l'ensemble des valeurs possibles. Par exemple, [61] a montré que le fait de savoir que les crises cardiaques surviennent à un taux réduit chez les patients japonais pouvait servir à réduire la gamme de valeurs pour un attribut sensible de la maladie d'un patient. Dans notre proposition, nous envisageons l'utilisation d'un paradigme dynamique de K-anonymat en fonction des objectifs de protection de la vie privée de chaque groupe de données.

4.4 Contexte

Les données de suivi du réseau fournissent des informations précieuses qui contribuent à la modélisation des comportements réseaux, à la défense contre les attaques du réseau et au développement de nouveaux protocoles. Ainsi, la diffusion des données d'une trace de réseau est très demandée par les chercheurs et les organisations pour promouvoir le développement des technologies de réseau. Toutefois, en raison de la nature sensible des données de suivi du réseau, les organisations risquent de publier ou de partager les données d'origine, ce qui pourrait exposer leur vie privée et celle de leurs clients au sein de leurs réseaux.

Plusieurs méthodes de défense des attaques de traces de réseau telles que l'empreinte et l'injection statiques ont été proposées, malheureusement, elles ne sont pas suffisantes pour protéger la vie privée des utilisateurs car la correspondance entre les adresses IP source et destination peut également aider l'adversaire à identifier l'hôte cible.

Nous examinons tout d'abord la décision du problème de l'obtention d'un K-anonymat privé optimal en analysant le niveau de risque pour la vie privée. Dans ce qui suit, nous présentons notre modélisation de la vie privée basée sur le mécanisme d'évaluation et de l'anonymat de K-Anonymat. Nous tenons compte de différentes hypothèses concernant les connaissances externes dont dispose un adversaire, comme indiqué dans notre modèle de menace et dans l'évaluation qualitative. Notre méthodologie est entièrement adaptée à la structure de données NetFlow présentée dans la figure 4.2.

Nous devons classer les données NetFlow en fonction de leur niveau de criticité de la vie privée. Nous pouvons évaluer qualitativement le niveau de la vie privée en évaluant

la relation entre les données et l'identité de l'émetteur. Dans notre cas, nous définissons quatre niveaux : données très critiques, données critiques, données Insensible et données publiques. La phase d'évaluation du niveau de privacy est l'entrée d'un processus dynamique de K-anonymat. En fonction de chaque niveau des données, nous pouvons définir les groupes de quasi-identifiants et leurs valeurs connexes de K.

4.5 Proposition

Dans cette section, nous formalisons l'approche de risque de contexte et de protection de la vie privée. Nous commençons par identifier les exigences en matière de privacy et de sécurité et nous examinons la phase d'analyse des risques privacy. Tout d'abord, nous procédons à une classification des données en fonction du niveau de leur criticité. Cela correspond à la classification des adresses IP, des ports et toutes les données consignées en groupes afin de fixer le niveau d'anonymat K correspondant. L'étape suivante consiste à utiliser un outil d'anonymisation pour appliquer la fonction d'anonymisation appropriée au champ approprié. Nous considérons qu'un datacenter collecte toutes les données enregistrées de Netflow auprès des différentes entreprises. L'objectif est d'analyser de façon anonyme les données enregistrées pour récupérer des informations utiles, telles que le pourcentage de connexions aux sites de médias sociaux. Notre travail permet de générer des statistiques de haute qualité par le biais de requêtes directes sur les bases de données sans révéler d'informations personnelles sensibles à partir du dataset.

4.5.1 Définition du niveau de confidentialité

Les approches d'anonymisation existantes se focalisent sur l'ajout d'une faible quantité de bruit (e.g. Gaussian Noise). Dans notre travail, nous visons à utiliser le bruit aléatoire en fonction du niveau de privacy des données groupées. Chaque groupe a un niveau de privacy défini. Nous choisissons la quantité de bruit en fonction du niveau de privacy ou de sensibilité des données.

4.5.2 Modélisation

Soit $T(A_1, ...A_n)$ est l'ensemble des attributs et QI_T les quasi-identifiants associés. QI_T est un ensemble d'attributs $(A_i, ...A_j) \subseteq (A_1, ...A_n)$.

D'après l'évaluation précédente de la privacy, nous pouvons diviser l'ensemble des quasi-identifiants en sous-groupes.

— Le QI_{TPr} est l'ensemble de quasi-identifiants associés à des données très critiques en termes de privacy.

— Le QI_{TM} est l'ensemble de quasi-identifiants associés aux données critiques moyennes en termes de privacy.

— Le QI_{TPu} est l'ensemble de quasi-identifiants associés aux données publiques en termes de privacy.

Cette approche peut être généralisée comme suit.

Definition 1 : Soit $QI_T(A_i, ..., A_k)$ l'ensemble des données critiques à anonymiser et QI_{PL} le sous-groupe de ces données critiques avec PL comme niveau de privacy. Le caractère aléatoire du bruit introduit par K-Anonymity est dérivé du niveau de privacy.

Après avoir défini différents groupes de quasi-identificateurs, nous nous concentrons sur la hiérarchie de généralisation du domaine. Nous attribuons à chaque QI_{PL} une hiérarchie de domaine. Nous définissons notre hiérarchie de généralisation en fonction du niveau de confidentialité des données anonymisées. Une distinction est faite entre les fonctions d'anonymisation basées sur le niveau de confidentialité des données. Dans le cas d'adresses IP dans les journaux NetFlow, nous pouvons considérer deux groupes principaux, le groupe public et le groupe privé. La figure 4.1 illustre les deux domaines D_{IP} : adresses IP du domaine et D_{Pr} : les ports de domaine et les hiérarchies de généralisation de valeur pour les deux domaines les plus importants dans le cas de NetFlow.

K a une valeur dynamique en fonction du niveau de confidentialité des données. Nous considérons notre table privée combinant des groupes d'attributs avec leur niveau privé, la valeur K-anonymity et les fonctions d'anonymisation correspondantes. Comme présenté par Sweeney dans [63], $D_i = dom(A_i, PT)$ désigne le domaine associé à l'attribut A_i dans la table privée PT.

4.5.3 Généralisation dynamique

Le concept de généralisation d'un attribut est assez simple ; une valeur est remplacée par une valeur moins spécifique, plus générale et fidèle à l'original. Étant donné le tableau PT, la généralisation peut être efficace pour produire un tableau RT basé sur PT. Le nombre de valeurs distinctes associées à chaque attribut n'augmente pas, de sorte que la substitution a tendance à mapper les valeurs au même résultat, ce qui peut réduire le nombre de tuples distincts dans RT.

Afin de distinguer le niveau de privacy des fonctions de pseudonymisation, nous définissons la fonction de généralisation comme $f_k i(t)$ sur tuple t, $f_k i$:, où $f_k i : A_1, ..., A_n \rightarrow f_k i(A_1), ..., f_k i(A_n)$ et i est le niveau de privacy variable. La classification des fonctions d'anonymisation dépend du niveau de privacy des données à appliquer.

$$\begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ . \\ . \\ . \\ . \\ A_n \end{pmatrix} \xrightarrow{f_j(A_i)} \begin{pmatrix} f_0(A_1) \\ f_1(A_2) \\ f_2(A_3) \\ f_k(.) \\ f_k(.) \\ f_l(.) \\ f_l(.) \\ f_l(A_n) \end{pmatrix}$$

Definition 2 :

Soit $Tk_i(A_1, ..., A_n)$ et $Tk_j(A_1, ..., A_n)$ deux tables définies sur le même ensemble d'attributs. Soit avec le même ensemble d'attributs, k_i pour le niveau de privacy dynamique. Pour chaque groupe de privacy, nous affectons un groupe de données et considérons la correspondance bijective entre deux tables.

1. $|Tk_i| = |Tk_j|$, k_i est le niveau de privacy

2. $\forall z = 1, ..., n : dom(Ak_z, Tk_i) \preceq_D dom(Ak_z, Tk_i)$
3. Il est possible de définir une correspondance bijective entre Tk_i et Tk_j qui associe chaque tuple t_i et t_j tel que $t_{i[Ak_z]} \preceq t_{j[A_z]}$.

Nous examinons la correspondance bijective entre les quasi-identificateurs regroupés par niveau de privacy et les éléments de généralisation. Dans les modèles de correspondance statique, la fonction de valeur de correspondance est exogène. La valeur de toute correspondance dépend à la fois de la fonction de production statique et des valeurs de continuation.

Afin de satisfaire la condition d'anonymat K, nous associons des données de niveau critique de privacy à la valeur la plus élevée de K.

Dans notre cas, nous considérons la classification des adresses IP comme présentée à la figure 4.3 en adresses IP publiques et adresses IP privées. Pour les adresses IP publiques, nous avons trois sous-groupes : adresses internes, spéciales et externes. Pour les adresses internes, nous faisons référence aux adresses IP routables domestiques. Les adresses IP publiques externes sont des adresses IP routables non domestiques. Les adresses IP publiques spéciales sont les adresses IP connues comme DNS, Facebook, etc. En fonction de la criticité du niveau de privacy des données, nous affectons à chaque groupe une fonction de pseudonymisation pour générer les attributs de la généralisation.

4.6 Analyse de privacy

Afin de respecter le k-anonymat, et dans le cas de NetFlow (voir la figure 4.2), nous pouvons distinguer quatre niveaux de protection de la vie privée présentés dans le tableau 4.1.

Dans cette section, nous allons d'abord présenter brièvement Cisco NetFlow. Ensuite, nous décrirons nos expériences d'application de la méthode d'anonymisation Crypto-Pan étendue pour échantillonner les traces de Cisco NetFlow. Enfin, nous analyserons les résultats.

Les journaux NetFlow Cisco contiennent des enregistrements des flux unidirectionnels entre paires d'ordinateurs/ports à travers un point d'instrumentation (par exemple un routeur) sur un réseau. Ces enregistrements peuvent être exportés à partir des routeurs ou des logiciels tels que ARGUS ou NTOP. NetFlow est une riche source d'informations pour l'analyse du trafic, comprenant certains ou la totalité des champs selon la version et la configuration : Paires d'adresses IP (source/destination), paires de ports (source/destination), protocole (TCP/UDP), nombre des paquets par seconde, timbres temporels (début/fin et/ou durée) et nombre d'octets. Nous configurons le logiciel pour travailler directement sur les données binaires des logs Cisco NetFlow. Nous aurions pu simplement compiler le code avec quelques modifications pour prendre une adresse IP décimale en pointillés via STDIN avec la clé comme argument. Il enverrait ensuite l'adresse IP anonyme à STDOUT. Un script perl peut analyser le journal et appeler ce binaire C++ pour effectuer l'anonymisation sur chaque adresse IP. Ensuite, nous appliquons la transformation K-anonymat.

Après la classification des données en fonction de leurs niveaux de privacy et la création des groupes, nous passons à la généralisation. Nous considérons que le groupe de quasi-identificateurs à son propre niveau de privacy. Les données très sensibles, telles que les

Champs	Niveau de privacy	La valeur k de l'anonymisation	Fonctions
Toutes les adresses	Très sensible	Valeur la plus élevée	Fonction de Hashage ou de chiffrement
Ports src et dst	Sensible	Valeur moyenne	Troncation ou aléatoire ou permutation
Masque source IP	Insensible	1	Permutation
Masque destination IP	Insensible	1	Permutation
Autres données	Publique	0	Pas de fonctions

TABLE 4.1 – les niveaux de privacy

adresses IP source et destination, fournissent des informations sensibles sur les entrées et les sorties et peuvent aider un traqueur à chercher des exemples de navigation sur Internet pour un utilisateur du réseau. Les ports sources et destination font également partie d'un groupe de données très sensibles. Le deuxième niveau de confidentialité est celui des données sensibles moyennes, qui concernent des données telles que les ID du réseau privé virtuel de source et de destination.

Le masque de source IP et le masque de destination IP peuvent se trouver dans des données non sensibles et le reste des données peut être considéré comme une information publique. En conséquence, nous pourrions attribuer la valeur 4 pour le groupe très sensible, la valeur 3 pour les données sensibles et la valeur 2 pour les données non sensibles : les autres données dont nous n'avons pas besoin lors de l'utilisation de K-anonymity. La même analyse pourrait être effectuée lors du choix de la fonction correspondante pour chaque groupe des données. Pour les données très sensibles, nous utilisons la fonction de hachage tronquée. Pour les données sensibles, nous utilisons la permutation.

Les résultats de cette simulation sont illustrés aux figures 4.4,4.5, 4.6 et 4.7. Dans tous les cas, nous pouvons voir la différence de partition des données avant et après la mise en œuvre de notre proposition d'anonymisation. En général, les plages des données ont changé lors de l'application de l'anonymat K dynamique. Une analyse plus approfondie pourrait être effectuée en affinant la définition d'un plus grand nombre de groupes de données. Par exemple, pour les adresses IP, nous pouvons considérer deux groupes pour les adresses publiques et privées. Un raffinement et une k-anonymisation plus optimale feront l'objet de travaux futurs.

4.7 Conclusion

Afin d'éviter l'obstacle du partage de journaux entre plusieurs utilisateurs, des techniques d'anonymisation fortes et efficaces sont nécessaires. Dans le présent chapitre, nous avons proposé un paradigme d'anonymisation dynamique en utilisant l'évaluation des risques liés à la protection de la vie privée. Le schéma classe les données en fonction de leurs niveaux de privacy. En fonction de ce niveau, nous fixons le K pour K-anonymat pour chaque groupe de données. La fonction de généralisation est également choisie en fonction du niveau de privacy des données. Le prototype implémenté a été testé avec des traces NetFlow. Nous identifions plusieurs niveaux de privacy critique pour classer le journal de NetFlow. Les résultats de la simulation montrent que la partition des données change complètement en appliquant un schéma K-anonymat optimal et dynamique.

1	date	time	router_ip	sampling	src_ip	dst_ip	nexthop	input	output	pkts	bytes	first	last		prot	sport	dport	flags	tos	src_as	dst_as
2	17/08/201	0:00	68.148.39.	1	68.130.17	157.0.175.4	0.0.0.0	76	326	2	160	1502899218		1502899218	17	0	0	0	0	0	0
3	17/08/201	0:00	68.148.39.	1	68.131.16	208.230.71	0.0.0.0	36	43	2	50	1502899233		1502899233	6	33311	80	16	0	0	0
4	17/08/201	0:00	68.148.39.	1	196.166.2	1240.172.7	0.0.0.0	322	326	2	181	1502899248		1502899248	6	60167	443	24	0	0	0
5	17/08/201	0:00	68.148.39.	1	2.41.205.1	68.129.53.	0.0.0.0	322	225	2	1133	1502899248		1502899248	6	443	63453	24	0	0	0
6	17/08/201	0:00	68.148.39.	1	68.129.93.	139.9.212.	0.0.0.0	217	393	2	162	1502899256		1502899256	50	0	0	0	0	0	0
7	17/08/201	0:01	68.148.39.	1	213.160.1	4138.80.141	0.0.0.0	244	42	11	15246	1502899205		1502899262	6	4666	60000	24	0	0	0
8	17/08/201	0:01	68.148.39.	1	72.185.0.1	203.117.2	0.0.0.0	222	182	2	69	1502899272		1502899272	6	48278	8555	2	0	0	0

FIGURE 4.2 – Journal NetFlow

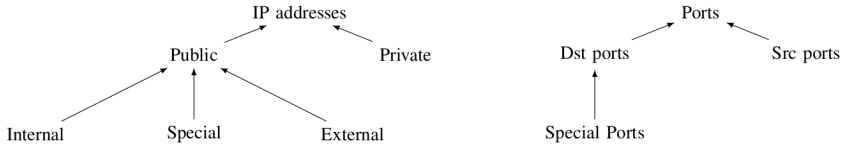
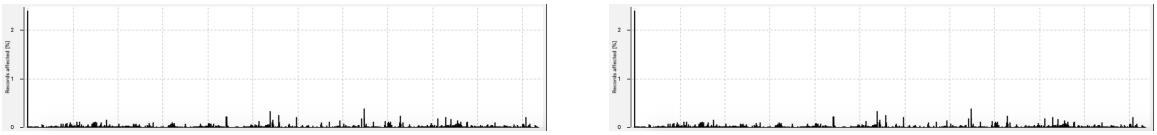
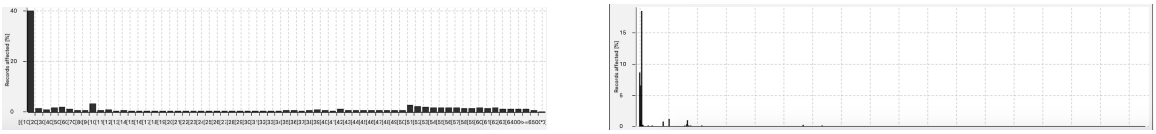


FIGURE 4.3 – Généralisation des domaines et des valeurs



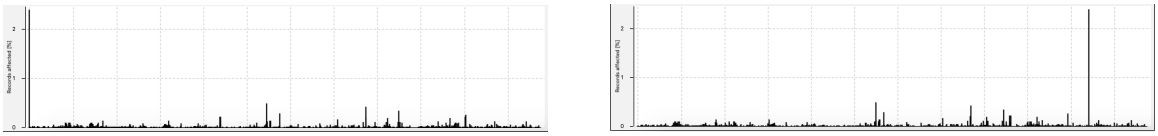
(a) Sans anomysation (b) Avec anomysation

FIGURE 4.4 – Anonymisation des sources IP



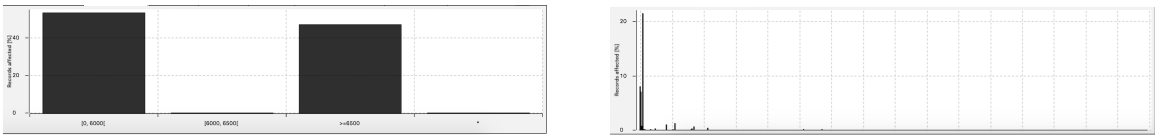
(a) Sans anomysation (b) Avec anomysation

FIGURE 4.5 – Anonymisation des ports sources



(a) Sans anomysation (b) Avec anomysation

FIGURE 4.6 – Anonymisation des destinations IP



(a) Sans anomysation (b) Avec anomysation

FIGURE 4.7 – Anonymisation des ports de destination

Conclusion et perspectives

Ce travail a permis de relever plusieurs défis liés à la préservation de la vie privée et au contrôle d'intégrité des données sensibles échangées entre plusieurs acteurs. La divulgation de ces données même dans un cadre d'application correcte des politiques de contrôle d'accès, représentent un risque particulièrement dangereux surtout avec les consignes strictes RGPD.

En outre, étant donné que les systèmes distribués hétérogènes basiques ne sont pas fiables, des techniques cryptographiques et d'anonymisation doivent être ajoutées pour garantir les propriétés de sécurité et de protection de la vie privée. Dans ce contexte, nous avons proposé deux solutions efficaces pour assurer l'échange et l'analyse des données privées stockées dans différents systèmes, ainsi que le cas des données stockées sur un seul nœud.

4.7.1 État de l'art

Nous avons présenté en détail les solutions existantes pour la protection de la vie privée pour les données externalisées.

4.7.2 Contributions

Le protocole Preserving Privacy in Distributed System -PPDS- et l' Analyse de sécurité

Nous avons d'abord abordé le problème de la préservation de la vie privée des données, stockées sur plusieurs nœuds. Nous avons développé un nouveau protocole, appelé PPDS conçu pour échanger et agréger des données sensibles sans intervention d'un tiers de confiance. PPDS définit un algorithme qui s'exécute sur des données issues de plusieurs nœuds et renvoie le résultat de croisement aux nœuds participants.

Nous avons validé notre protocole formellement et par une expérimentation sur des données bancaires. Nous avons analysé les temps de traitement et de réponse de ce protocole. Les résultats expérimentaux montrent d'excellents gains de performance et une bonne optimisation du temps de calcul. Nous avons analysé la sécurité de notre protocole via une analyse de sécurité de l'attaque Man-In-The-Midlle comme exemple et nous avons proposé une solution basée sur l'entropie pour détecter les attaques de type DoS.

Privacy as Service

L'anonymisation efficace est la clé pour l'analyse des données massives tout en préservant la vie privée des utilisateurs. Une entreprise doit être en mesure de partager et consolider les données qu'elle recueille dans ses services et dans son réseau. La prévention de la réidentification révèle également d'une importance particulière. Le but principal de cette solution est de fournir une amélioration d'un concept original d'anonymisation des données afin de rendre impossible la réidentification des utilisateurs. Nous appliquons notre solution sur un Log NetFlow pour validation. La solution inclut un processus d'analyse des risques lié à la protection de la vie privée pour classer les données en fonction de leur niveau de criticité. Nous introduisons un paradigme dynamique de K-anonymat qui tient compte des résultats de l'évaluation des risques liés à la criticité des identifiants. Enfin, nous évaluons empiriquement la performance et la répartition des données avec la solution proposée.

4.7.3 Les perspectives

Les résultats de contribution présentés dans cette thèse ouvrent de nouvelles perspectives dans le domaine de la préservation de la vie privée que nous avons exploré. Nous décrivons quelques directions de recherche potentiellement intéressantes qui découlent directement de notre travail.

Pour le protocole PPDS : Ce dernier considère des nœuds semi-honnêtes, c'est-à-dire qu'il suit les règles du protocole mais essaie d'apprendre autant d'informations que possible sur les données sensibles des autres nœuds. Nous aimerons dans le futur ajuster notre protocole à un autre contexte : Une autre structure où les nœuds sont malveillants et sont autorisés à effectuer des opérations sur les données sensibles et privées. On cite ici, le cas de partage et d'agrégation de données médicales : ces deux opérations sont essentielles dans la pratique clinique, l'assurance-maladie et la recherche médicale moderne. Le défi consiste à effectuer un partage de données qui préserve la confidentialité individuelle et l'utilité des données. Les lacunes des technologies traditionnelles de protection de la vie privée signifient que les institutions s'appuient sur des contrats de partage de données sur-mesure. La longueur du processus et de l'administration induite par ces contrats augmente l'inefficacité du partage des données et peut décourager d'importants traitements cliniques et recherches médicales.

Pour l'anonymisation des données : Notre travail pourrait être étendu par une étude comparative entre les différentes techniques d'anonymisation en ajoutant une phase d'analyse des risques pour la classification des données critiques afin d'évaluer les risques liés à la protection de la vie privée. En fonction de ce niveau, nous fixons le facteur d'anonymisation pour chaque groupe de données.

Bibliographie

[1] P. R. Clearinghouse, "Privacy rights clearinghouse," *Retrieved June*, vol. 1, p. 2016, 2015.

[2] L. Sweeney, "k-anonymity : A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[3] S. K. Akinade, "Database as a service : Security and privacy issues, and appropriate controls," 2020.

[4] "Distributed systems security issue." https://www.cybok.org/media/downloads/Distributed_Systems_Security_issue_1.0.pdf.

[5] J. Wu, *Distributed system design*. CRC press, 1998.

[6] J. Bacon and K. Moody, "Access control in distributed systems," in *Computer Systems*, pp. 21–28, Springer, 2004.

[7] D. Malkhi and M. K. Reiter, "An architecture for survivable coordination in large distributed systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 2, pp. 187–202, 2000.

[8] S. K. Rahimi and F. S. Haug, *Distributed database management systems : A Practical Approach*. John Wiley & Sons, 2010.

[9] M. van Steen and A. S. Tanenbaum, "A brief introduction to distributed systems," *Computing*, vol. 98, no. 10, pp. 967–1009, 2016.

[10] B. Hamid and D. Weber, "Engineering secure systems : Models, patterns and empirical validation," *Computers & Security*, vol. 77, pp. 315–348, 2018.

[11] Z. Tbatou, A. Asimi, Y. Asimi, Y. Sadqi, and A. Guezzaz, "A new mutuel kerberos authentication protocol for distributed systems.," *IJ Network Security*, vol. 19, no. 6, pp. 889–898, 2017.

[12] A. S. Tanenbaum and M. Van Steen, *Distributed systems : principles and paradigms*. Prentice-hall, 2007.

[13] M. Van Steen and A. S. Tanenbaum, *Distributed systems*. Maarten van Steen Leiden, The Netherlands, 2017.

[14] G. Joy Persial, M. Prabhu, and R. Shanmugalakshmi, "Side channel attack-survey," *Int J Adva Sci Res Rev*, vol. 1, no. 4, pp. 54–57, 2011.

[15] Y. Saito and M. Shapiro, "Optimistic replication," *ACM Computing Surveys (CSUR)*, vol. 37, no. 1, pp. 42–81, 2005.

[16] X. Zhang, C. Liu, S. Nepal, C. Yang, and J. Chen, "Privacy preservation over big data in cloud systems," in *Security, Privacy and Trust in Cloud Systems*, pp. 239–257, Springer, 2014.

[17] M. Van Dijk and A. Juels, "On the impossibility of cryptography alone for privacy-preserving cloud computing.," *HotSec*, vol. 10, pp. 1–8, 2010.

[18] J. Sedayao and I. I. Enterprise Architect, "Enhancing cloud security using data anonymization," *White Paper, Intel Coporation*, 2012.

[19] N. Suri, "Distributed systems security knowledge area," 2019.

[20] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE transactions on dependable and secure computing*, vol. 1, no. 1, pp. 11–33, 2004.

[21] T. Isdal, M. Piatek, A. Krishnamurthy, and T. Anderson, "Privacy-preserving p2p data sharing with oneswarm," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 4, pp. 111–122, 2010.

[22] J. Seibert, X. Sun, C. Nita-Rotaru, and S. Rao, "Towards securing data delivery in peer-to-peer streaming," in *2010 Second International Conference on COMmunication Systems and NETworks (COMSNETS 2010)*, pp. 1–10, IEEE, 2010.

[23] R. B. de Almeida, J. A. M. Natif, A. P. C. da Silva, and A. B. Vieira, "Pollution and whitewashing attacks in a p2p live streaming system : analysis and counter-attack," in *2013 IEEE International Conference on Communications (ICC)*, pp. 2006–2010, IEEE, 2013.

[24] J. Liang, N. Naoumov, and K. W. Ross, "The index poisoning attack in p2p file sharing systems.," in *INFOCOM*, pp. 1–12, Citeseer, 2006.

[25] N. Naoumov and K. Ross, "Exploiting p2p systems for ddos attacks," in *Proceedings of the 1st international conference on Scalable information systems*, pp. 47–es, 2006.

[26] A. Walters, D. Zage, and C. N. Rotaru, "A framework for mitigating attacks against measurement-based adaptation mechanisms in unstructured multicast overlay networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 6, pp. 1434–1446, 2008.

[27] D. Li, J. Wu, and Y. Cui, "Defending against buffer map cheating in donet-like p2p streaming," *IEEE Transactions on Multimedia*, vol. 11, no. 3, pp. 535–542, 2009.

[28] J. R. Douceur, "The sybil attack," in *International workshop on peer-to-peer systems*, pp. 251–260, Springer, 2002.

[29] A. Singh *et al.*, "Eclipse attacks on overlay networks : Threats and defenses," in *In IEEE INFOCOM*, Citeseer, 2006.

[30] N. Suri, "Distributed systems security knowledge area issue.,"

[31] F. DePaoli and L. Mariani, "Dependability in peer-to-peer systems," *IEEE Internet Computing*, vol. 8, no. 4, pp. 54–61, 2004.

[32] G. Gheorghe, R. L. Cigno, and A. Montresor, "Security and privacy issues in p2p streaming systems : A survey," *Peer-to-Peer Networking and Applications*, vol. 4, no. 2, pp. 75–91, 2011.

[33] G. Urdaneta, G. Pierre, and M. V. Steen, "A survey of dht security techniques," *ACM Computing Surveys (CSUR)*, vol. 43, no. 2, pp. 1–49, 2011.

[34] Y.-K. Kwok, "Autonomic peer-to-peer systems : incentive and security issues," in *Autonomic Computing and Networking*, pp. 205–236, Springer, 2009.

[35] S. Androutsellis-Theotokis and D. Spinellis, "A survey of peer-to-peer content distribution technologies," *ACM computing surveys (CSUR)*, vol. 36, no. 4, pp. 335–371, 2004.

[36] D. S. Wallach, "A survey of peer-to-peer security issues," in *International symposium on software security*, pp. 42–57, Springer, 2002.

[37] D. J. Abadi, D. Carney, U. Cetintemel, M. Cherniack, C. Convey, S. Lee, M. Stonebraker, N. Tatbul, and S. Zdonik, "Aurora : a new model and architecture for data stream management," *the VLDB Journal*, vol. 12, no. 2, pp. 120–139, 2003.

[38] K. Kolomvatsos, C. Anagnostopoulos, and S. Hadjiefthymiades, "An efficient time optimized scheme for progressive analytics in big data," *Big Data Research*, vol. 2, no. 4, pp. 155–165, 2015.

[39] M. James, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. Byers, "The next frontier for innovation, competition, and productivity," *Big data*, 2011.

[40] J. Gantz and D. Reinsel, "Extracting value from chaos," *IDC iview*, vol. 1142, no. 2011, pp. 1–12, 2011.

[41] M. Chen, S. Mao, and Y. Liu, "Big data : A survey," *Mobile networks and applications*, vol. 19, no. 2, pp. 171–209, 2014.

[42] C.-W. Tsai, C.-F. Lai, H.-C. Chao, and A. V. Vasilakos, "Big data analytics : a survey," *Journal of Big data*, vol. 2, no. 1, pp. 1–32, 2015.

[43] A. Mehmood, I. Natgunanathan, Y. Xiang, G. Hua, and S. Guo, "Protection of big data privacy," *IEEE access*, vol. 4, pp. 1821–1834, 2016.

[44] P. Jain, N. Pathak, P. Tapashetti, and A. Umesh, "Privacy preserving processing of data decision tree based on sample selection and singular value decomposition," in *2013 9th international conference on information assurance and security (IAS)*, pp. 91–95, IEEE, 2013.

[45] C. C. Aggarwal, N. Ashish, and A. Sheth, "The internet of things : A survey from the data-centric perspective," in *Managing and mining sensor data*, pp. 383–428, Springer, 2013.

[46] P. Porambage, M. Ylianttila, C. Schmitt, P. Kumar, A. Gurtov, and A. V. Vasilakos, "The quest for privacy in the internet of things," *IEEE Cloud Computing*, vol. 3, no. 2, pp. 36–45, 2016.

[47] J. Han, M. Ishii, and H. Makino, "A hadoop performance model for multi-rack clusters," in *2013 5th International Conference on Computer Science and Information Technology*, pp. 265–274, IEEE, 2013.

[48] M. Gudipati, S. Rao, N. D. Mohan, and N. K. Gajja, "Big data : Testing approach to overcome quality challenges," *Big Data : Challenges and Opportunities*, vol. 11, no. 1, pp. 65–72, 2013.

[49] L. Xu, C. Jiang, J. Wang, J. Yuan, and Y. Ren, "Information security in big data : privacy and data mining," *Ieee Access*, vol. 2, pp. 1149–1176, 2014.

[50] S. Liu, "Exploring the future of computing," *IT Professional*, vol. 15, no. 1, pp. 2–3, 2013.

[51] M. Sokolova and S. Matwin, "Personal privacy protection in time of big data," in *Challenges in computational statistics and data mining*, pp. 365–380, Springer, 2016.

[52] H. Cheng, C. Rong, K. Hwang, W. Wang, and Y. Li, "Secure big data storage and sharing scheme for cloud tenants," *China Communications*, vol. 12, no. 6, pp. 106–115, 2015.

[53] P. Mell, T. Grance, *et al.*, "The nist definition of cloud computing," 2011.

[54] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, and A. V. Vasilakos, "Security and privacy for storage and computation in cloud computing," *Information sciences*, vol. 258, pp. 371–386, 2014.

[55] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE communications surveys & tutorials*, vol. 15, no. 2, pp. 843–859, 2012.

[56] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *2010 proceedings ieee infocom*, pp. 1–9, Ieee, 2010.

[57] C. Liu, R. Ranjan, X. Zhang, C. Yang, D. Georgakopoulos, and J. Chen, "Public auditing for big data storage in cloud computing–a survey," in *2013 IEEE 16th International Conference on Computational Science and Engineering (CSE)*, pp. 1128–1135, IEEE Computer Society, 2013.

[58] C. Liu, J. Chen, L. T. Yang, X. Zhang, C. Yang, R. Ranjan, and R. Kotagiri, "Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 9, pp. 2234–2244, 2013.

[59] K. Xu, H. Yue, L. Guo, Y. Guo, and Y. Fang, "Privacy-preserving machine learning algorithms for big data systems," in *2015 IEEE 35th international conference on distributed computing systems*, pp. 318–327, IEEE, 2015.

[60] Y. Zhang, T. Cao, S. Li, X. Tian, L. Yuan, H. Jia, and A. V. Vasilakos, "Parallel processing systems for big data : a survey," *Proceedings of the IEEE*, vol. 104, no. 11, pp. 2114–2136, 2016.

[61] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness : Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115, IEEE, 2007.

[62] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity : Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, pp. 3–es, 2007.

[63] P. Samarati, "Protecting respondents privacy in microdata release," *IEEE transaction on Knowledge and Data Engineering*, vol. 13, no. 6, 2001.

[64] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information : k-anonymity and its enforcement through generalization and suppression," 1998.

[65] P. Jain, M. Gyanchandani, and N. Khare, "Big data privacy : a technological perspective and review," *Journal of Big Data*, vol. 3, no. 1, pp. 1–25, 2016.

[66] R. Wang, Y. Zhu, T.-S. Chen, and C.-C. Chang, "Privacy-preserving algorithms for multiple sensitive attributes satisfying t-closeness," *Journal of Computer Science and Technology*, vol. 33, no. 6, pp. 1231–1242, 2018.

[67] S. Y. Ko, K. Jeon, and R. Morales, "The hybrex model for confidentiality and privacy in cloud computing.," *HotCloud*, vol. 11, pp. 8–8, 2011.

[68] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," *IEEE Network*, vol. 28, no. 4, pp. 46–50, 2014.

[69] C. Dwork, "Differential privacy," in *International Colloquium on Automata, Languages, and Programming*, pp. 1–12, Springer, 2006.

[70] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "Trac : Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*, pp. 1231–1239, IEEE, 2014.

[71] H. Zakerzadeh, C. C. Aggarwal, and K. Barker, "Privacy-preserving big data publishing," in *Proceedings of the 27th international conference on scientific and statistical database management*, pp. 1–11, 2015.

[72] J. Dean and S. Ghemawat, "Mapreduce : simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[73] R. Lämmel, "Google's mapreduce programming model—revisited," *Science of computer programming*, vol. 70, no. 1, pp. 1–30, 2008.

[74] Z. Yan, W. Ding, V. Niemi, and A. V. Vasilakos, "Two schemes of privacy-preserving trust evaluation," *Future Generation Computer Systems*, vol. 62, pp. 175–189, 2016.

[75] Y. Zhang, S. Fong, J. Fiaidhi, and S. Mohammed, "Real-time clinical decision support system with data stream mining," *Journal of biomedicine and biotechnology*, vol. 2012, 2012.

[76] E. Mohammadian, M. Noferesti, and R. Jalili, "Fast : fast anonymization of big data streams," in *Proceedings of the 2014 International Conference on Big Data Science and Computing*, pp. 1–8, 2014.

[77] A. Forestiero, "Fads : Flocking anomalies in data streams," in *2012 6th IEEE International Conference Intelligent Systems*, pp. 461–466, IEEE, 2012.

[78] T. Haferlach, A. Kohlmann, L. Wieczorek, G. Basso, G. Te Kronnie, M.-C. Béné, J. De Vos, J. M. Hernández, W.-K. Hofmann, K. I. Mills, *et al.*, "Clinical utility of microarray-based gene expression profiling in the diagnosis and subclassification of leukemia : report from the international microarray innovations in leukemia study group," *Journal of Clinical Oncology*, vol. 28, no. 15, p. 2529, 2010.

[79] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, *et al.*, "Molecular classification of cancer : class discovery and class prediction by gene expression monitoring," *science*, vol. 286, no. 5439, pp. 531–537, 1999.

[80] B. Kayyali, D. Knott, and S. Van Kuiken, "The big-data revolution in us health care : Accelerating value and innovation," *Mc Kinsey & Company*, vol. 2, no. 8, pp. 1–13, 2013.

[81] P. Protection and A. C. Act, "Patient protection and affordable care act," *Public law*, vol. 111, no. 48, pp. 759–762, 2010.

[82] L. Marcotte, J. Seidman, K. Trudel, D. M. Berwick, D. Blumenthal, F. Mostashari, and S. H. Jain, "Achieving meaningful use of health information technology : a guide for physicians to the ehr incentive programs," *Archives of internal medicine*, vol. 172, no. 9, pp. 731–736, 2012.

[83] M. H. Tekieh and B. Raahemi, "Importance of data mining in healthcare : a survey," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pp. 1057–1062, 2015.

[84] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "Bsn : Boundary sensitive network for temporal action proposal generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018.

[85] M. G. Haselton, D. Nettle, and D. R. Murray, "The evolution of cognitive bias," *The handbook of evolutionary psychology*, pp. 1–20, 2015.

[86] K. Hill, "How target figured out a teen girl was pregnant before her father did," *Forbes, Inc*, 2012.

[87] M. Kantarcioğlu and C. Clifton, "Assuring privacy when big brother is watching," in *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pp. 88–93, 2003.

[88] O. Tene and J. Polonetsky, "Privacy in the age of big data : a time for big decisions," *Stan. L. Rev. Online*, vol. 64, p. 63, 2011.

[89] A. Lysyanskaya, R. L. Rivest, A. Sahai, and S. Wolf, "Pseudonym systems," in *International Workshop on Selected Areas in Cryptography*, pp. 184–199, Springer, 1999.

[90] L. Lu, J. Han, L. Hu, J. Huai, Y. Liu, and L. M. Ni, "Pseudo trust : Zero-knowledge based authentication in anonymous peer-to-peer protocols," in *2007 IEEE International Parallel and Distributed Processing Symposium*, pp. 1–10, IEEE, 2007.

[91] U. Feige, A. Fiat, and A. Shamir, "Zero-knowledge proofs of identity," *Journal of cryptology*, vol. 1, no. 2, pp. 77–94, 1988.

[92] J. Brandt, I. Damgård, P. Landrock, and T. Pedersen, "Zero-knowledge authentication scheme with secret key exchange," in *Conference on the Theory and Application of Cryptography*, pp. 583–588, Springer, 1988.

[93] T. Beth, "Efficient zero-knowledge identification scheme for smart cards," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 77–84, Springer, 1988.

[94] C. Shields and B. N. Levine, "A protocol for anonymous communication over the internet," in *Proceedings of the 7th ACM Conference on Computer and Communications Security*, pp. 33–42, 2000.

[95] W. Diffie and M. Hellman, "" new directions in cryptography" ieee transactions on information theory, v. it-22, n. 6," 1976.

[96] E. Barker, E. Barker, W. Burr, W. Polk, M. Smid, *et al.*, *Recommendation for key management : Part 1 : General*. National Institute of Standards and Technology, Technology Administration, 2006.

[97] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, *et al.*, "The avispa tool for the automated validation of internet security protocols and applications," in *International conference on computer aided verification*, pp. 281–285, Springer, 2005.

[98] A. Mallik, "Man-in-the-middle-attack : Understanding in simple words," *Cyberspace : Jurnal Pendidikan Teknologi Informasi*, vol. 2, no. 2, pp. 109–134, 2019.

[99] R. H. Jhaveri, S. J. Patel, and D. C. Jinwala, "Dos attacks in mobile ad hoc networks : A survey," in *2012 second international conference on advanced computing & communication technologies*, pp. 535–541, IEEE, 2012.

[100] N. Hamza, H. A. Hefny, *et al.*, "Attacks on anonymization-based privacy-preserving : a survey for data mining and data publishing," 2013.

[101] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 571–588, 2002.

[102] T. NL and V. t Closeness, "Privacy beyond k-anonymity and l-diversity," ICDE, 2007.