REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABOU-BEKR BELKAID - TLEMCEN

THÈSE

Présentée à :

FACULTE DES SCIENCES – DEPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

DOCTORAT EN SCIENCES

Spécialité: *Informatique*

Par :

Mr Belabed Amine

Sur le thème

La protection de la vie privée sur Internet

Soutenue publiquement le **28 Juin 2018** à Tlemcen devant le jury composé de :

Mr Benamar Abdelkarim	MCA	Université de Tlemcen	Président
Mr Chikh Mohammed Amine	Professeur	Université de Tlemcen	Directeur de thèse
M^{me} Aïmeur Esma	Professeur	Université de Montreal, Canada	Co-Directeur de thèse
M^{me} Labraoui Nabila	MCA	Université de Tlemcen	Examinatrice
Mr Amar Bensaber Djamel	MCA	ESI, Sidi Bel Abbès	Examinateur
Mr Toumouh Adil	MCA	Université de Sidi Bel Abbès	Examinateur
Mr Saïs Lakhdar	Professeur	Université d'Artois, France	Invité
Mr Hadjila Fethallah	MCB	Université de Tlemcen	Invité

Laboratoire de Recherche en Informatique de Tlemcen (LRIT)
BP 119, 13000 Tlemcen - Algérie

À la mémoire de mon père,
à ma mère,
à ma femme,
à mon fils Iyad,
à toute ma famille. . .

Avant-propos

Remerciements

Tout d'abord, Je remercie Allah qui m'a donné la force pour accomplir ce modèste travail.

Je tiens à remercier en premier lieu, **Mr Chikh Mohammed Amine** et **Mme Aïmeur Esma**, mes directeurs de thèse pour leurs orientations, leurs encouragements, leurs conseils et leur sympathie qui m'ont permis de mener à bien cette thèse.

J'exprime ma profonde reconnaissance **Mr Sais Lakhder** professeur à l'université d'Artois, pour m'avoir accueilli durant 18 mois au laboratoire CRIL (Université d'Artois, France), pour son aide, pour sa sympathie et sa gentillesse.

Je suis très honoré par la présence de **Mr Benamar Abdelkarim**, qui a accepté de présider le jury de ma thèse.

J'adresse mes sincères remerciements à **Mme Labraoui Nabila**, **Mr Amar Bensaber Djamel** et **Mr Toumouh Adil**, qui ont accepté d'être les examinateurs de cette thèse. Qu'ils trouvent ici, mes plus vifs remerciements pour l'effort qu'ils ont fait pour lire mon manuscrit et l'intérêt qu'ils ont porté à mon travail.

Je remercie ma famille de m'avoir donné le courage d'accomplir cette thèse. J'adresse mes sincères remerciements à mes collègues : Mr Hadjila Fethallah et Mr Merzoug Mohammed, Pour leurs conseils et soutien continu durant cette thèse.

Enfin, je tiens à remercier qui, de près ou de loin, ont collaboré à l'aboutissement de ce travail.

Résumé

L'INTERNET devient un moyen de plus en plus indispensable, presque utilisable dans touts les domaines (communications, commerce, relations sociales, ...). La grande masse d'information qui circule sur Internet ouvre l'appétit de pas mal de gens et d'entreprises, qui utilisent ces informations hors de leur contexte légitime, touchant directement à la vie privée des personnes. Les conséquences de cette utilisation peuvent varier d'une simple collecte d'information qui touche à l'intimité des personnes, jusqu'au vol d'identité qui induit des conséquences graves et directes sur la victime, de ce fait, protéger sa vie privée n'est plus une question d'éthique.

Le but de notre thèse est de faire une étude sur la protection de la vie privée sur internet. Nous avons choisi de traiter cette problématique sur plusieurs niveaux : le niveau accès et présentation, le niveau API et services et enfin le niveau stockage et base de données. Ces niveaux modélisent le cycle de vie des données circulant sur Internet depuis leur divulgation par les utilisateurs jusqu'à leur utilisation et stockage par les fournisseurs de services.

Au niveau accès, nous avons traité la problématique de protection des données personnelles contre l'hameçonnage (le Phishing), le type d'attaque le plus utilisé sur Internet pour le vol d'identité. A ce niveau nous avons proposé une approche de lute contre le Phishing. Cette approche se base sur un filtre à deux niveaux : une liste blanche personnalisée et un classificateur SVM. Les pages de Phishing qui ne sont pas filtrées au niveau de la liste blanche sont traitées par le classificateur SVM, ce qui donne une meilleur efficacité.

Au niveau API et services, nous avons traité la protection de la vie privée dans les services Web. Ce choix est motivé par le rôle central que cette technologie joue dans le fonctionnement d'Internet. Dans ce contexte, nous avons proposé un Framework de sélection de services qui préserve les exigences de vie privée des utilisateurs ainsi que les fournisseurs de services. Pour ce faire, nous avons introduit un formalisme qui permet de modéliser le problème de sélection sous forme d'un problème d'optimisation, avec un objectif de minimiser la fonction de risque relatif à l'utilisation des données privées par les services sélectionnés.

Au niveau stockage, nous nous sommes concentrés sur l'anonymat des données collectées et stockées chez les fournisseurs de services. Même pour des raisons légitimes, les données collectées sont généralement partagées et publiées, ce qui expose les propriétaires de ces données à des risques d'attaques sur leur vie privée. Dans ce cadre nous avons proposé une approche de protection qui se base sur la publication des données fictives au lieu de vrais données. Les données fictives sont générées en utilisant des modèles issues des données originales, et en se basant sur les techniques de Machine Learning. Les données générées gardent certaines propriétés des données originales, ce qui assure à la fois une forte protection et une grande utilité.

Mots clés : Vie privée, Internet, Phishing, selection des services, données personnelles, anonymat.

Abstract

INternet is becoming more and more indispensable, almost usable in all fields (communications, commerce, social relations, etc.). The vast amount of information circulating on the Internet opens the appetite of many people and businesses, who use this information outside of their legitimate context, thus it affects the privacy of individuals. The consequences of this use can vary from a simple collection of information that affects the intimacy of individuals, to the identity theft that has serious and direct consequences for the victim. Thereby, protecting individual's privacy is no longer an ethical issue.

The aim of our thesis is to make a study on privacy protection in Internet. We have addressed several levels of this issue : the access and presentation level, the API and services level and finally the storage and database level. These levels model the life cycle of data circulating on the Internet from its disclosure by users to its use and storage by the service providers.

At the access level, we dealt with the problem of protecting personal data against phishing, the most common type of attack on the Internet for identity theft. At this level we have proposed an approach against Phishing. This approach is based on a two-level filter : a personlized whitelist and an SVM classifier. Phishing pages that are not filtered at the white list level are processed by the SVM classifier, which gives better efficiency.

At the API and services level, we have addressed the privacy protection in web services. This choice is motivated by the central role that this technology plays in the functioning of the Internet. In this context, we have proposed a Service Selection Framework that preserves the privacy requirements of users as well as service providers. To do this, we have introduced a formalism that allows to model the selection problem as an optimization problem, with an objective to minimize the risk function related to the use of private data by the selected services.

At the storage level, we focused on the anonymity of data collected and stored at service providers. Even for legitimate reasons, the data collected is generally shared and published, exposing data owners to the risk of attacks on their privacy. In this context, we have proposed a protection approach based on the publication of fictitious data instead of real data. Fictional data are generated using models from the original data, using Machine Learning techniques. The generated data keeps some properties of the original data, which ensures both a strong protection and a great utility.

Keywords :Privacy, Internet, Phishing, Service Selection, Personal data, Anonymity.

ملخص

الهدف من هذه الرسالة هو عمل دراسة حول حماية الخصوصية على الإنترنت. لقد اخترنا معالجة هذه المشكلة على عدة مستويات: مستوى الدخول إلى الخدمة ، مستوى واجهة برمجة التطبيقات والخدمات وأخيراً مستوى التخزين وقاعدة البيانات. هذه المستويات نموذج دورة حياة البيانات المتداولة على الإنترنت من الكشف عنها من قبل المستخدمين لاستخدامها وتخزينها من قبل مقدمي الخدمات. على مستوى الدخول إلى الخدمة ، تعاملنا مع مشكلة حماية البيانات الشخصية ضد التصيد الاحتيالي ، وهو النوع الأكثر شيوعًا للهجوم على الإنترنت لسرقة الهوية. على هذا المستوى ، اقترحنا مقاربة ضد التصيّد الاحتيالي. يعتمد هذا الأسلوب على عامل تصفية من مستويين: قائمة بيضاء مخصصة ومصنف SVM . تتم معالجة صفحات التصيد الاحتيالي التي لم تتم تصفيتها على مستوى القائمة البيضاء بواسطة مصنف SVM ، مما يعطي كفاءة أفضل. على مستوى واجهة برمجة التطبيقات (API)، قمنا بمعالجة حماية الخصوصية في خدمات الويب. الحافز من وراء هذا الاختيار هو الدور المركزي الذي تلعبه هذه التقنية في عمل الإنترنت. في هذا السياق ، اقترحنا إطار عمل لاختيار خدمات الويب ،هذا الاطار يحافظ على متطلبات الخصوصية للمستخدمين وكذلك مقدمي الخدمات. لهذا الهدف ، أدخلنا نموذج يسمح بنمذجة مشكلة الاختيار كمشكلة تحسين ، و هذا لهدف تقليل وظيفة المخاطرة المرتبطة باستخدام البيانات الخاصة بواسطة الخدمات المختارة. على مستوى التخزين ، ركزنا على عدم الكشف عن هوية البيانات التي تم جمعها وتخزينها من طرف مقدمي الخدمات. في هذا السياق ، اقترحنا منهجًا للحماية يستند إلى نشر بيانات مولدة بدلاً من البيانات الحقيقية. يتم إنشاء البيانات المولدة باستخدام نماذج من البيانات الأصلية ، باستخدام تقنيات التعلم الآلي. تحتفظ البيانات المولدة ببعض خصائص البيانات الأصلية ، مما يضمن حماية قوية و استخدام أمثل.

الكلمات المفتاحية: الخصوصية ، الإنترنت ، التصيد ، اختيار الخدمات ، البيانات الشخصية ، عدم الكشف عن الهوية.

Table des matières

Table des figures

Liste des tableaux

"If this is the age of information, then privacy is the issue of our times."

–Alessandro Acquisti

1

Introduction Générale

▷ *Introduction, Contexte, problématiques et contributions.* ◁

Plan du chapitre

1.1 Contexte

Dès son avènement, Internet a réussi à rassembler une grande partie des connaissances du monde dans un endroit centralisé que tout le monde peut utiliser. Après quelques décennies, cette technologie est devenue un enjeu majeur dans les communications, le commerce ainsi que les relations sociales. Les innovations récentes dans le domaine des technologies de l'information et des communications, y compris la disponibilité accrue de dispositifs intelligents (Smarphone, PDA,...) ainsi que les réseaux sans fil et de capteurs, ont rendu la masse d'information circulant sur Internet très importante. Cette information est devenue rapidement la nouvelle monnaie. Cela, a fait que l'Internet d'aujourd'hui est plein d'entreprises et d'organisations dont leur seul but est d'obtenir autant d'informations que possible. Les informations collectées peuvent être utilisées hors de leur contexte légitime, touchant directement à la vie privée des personnes. Les conséquences de cette utilisation peuvent varier d'une simple collecte d'informations qui touche à l'intimité des personnes, jusqu'au vol d'identité qui induit des conséquences graves et directes sur la victime. Les menaces sur la vie privée sont multiples et touchent les différentes phases de cycle de vie des données circulant sur Internet [Brands00].

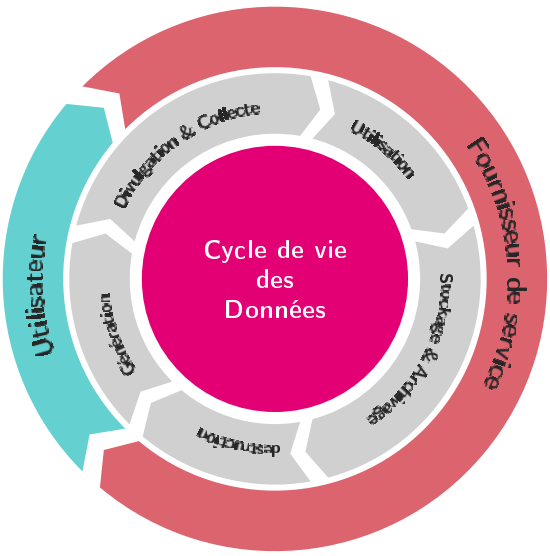


FIGURE 1.1 – Le cycle de vie des Données

Ce cycle de vie, comme la figure 1.1 montre, comporte plusieurs phases et évolue au niveau des utilisateurs, et des fournisseurs de services (ou collecteurs de données). Le cycle commence au niveau utilisateur, lorsque ce dernier *génère* des données personnelles d'une manière intentionnelle (ex. Historique de navigation,

achats sur internet, géolocalisation,...), ou non-intentionnelle (ex. Nom, prénom, date de naissance, ...). La phase *divulgation et collecte* commence au moment où les données générées passent au niveau fournisseurs de services. Ce passage peut être fait avec l'accord de l'utilisateur, si les données sont indispensables pour l'exécution de service (ex. Nom et prénom pour une réservation d'un vol ou d'un hôtel), ou sans son accord, lorsque le fournisseur de service collecte des données non nécessaires pour l'exécution de service et à l'insu de l'utilisateur (ex. l'historique des sites visités, ou des déplacement pour un service de navigation). Après la phase de génération et de divulgation, les données collectées passent par les phases *utilisation*, *stokage et archivage* et enfin la *destruction*. Les diverses phases du cycle de vie de données ne sont pas forcément indépendantes et séquentielles. Par exemple, la génération des données peut se faire durant l'utilisation d'un service. De la même façon, la phase de l'utilisation des données peut durer le long de la phase de stockage et jusqu'à la destruction des données. Notons que la phase *utilisation* se définit par plusieurs actions, parmi lesquelles, on cite : *le partage*, *la transformation* et *la diffusion*.

À chaque phase du cycle de vie des données, se révèlent des menaces et des problèmes relatifs à la vie privée des utilisateurs (voir figure 1.2).

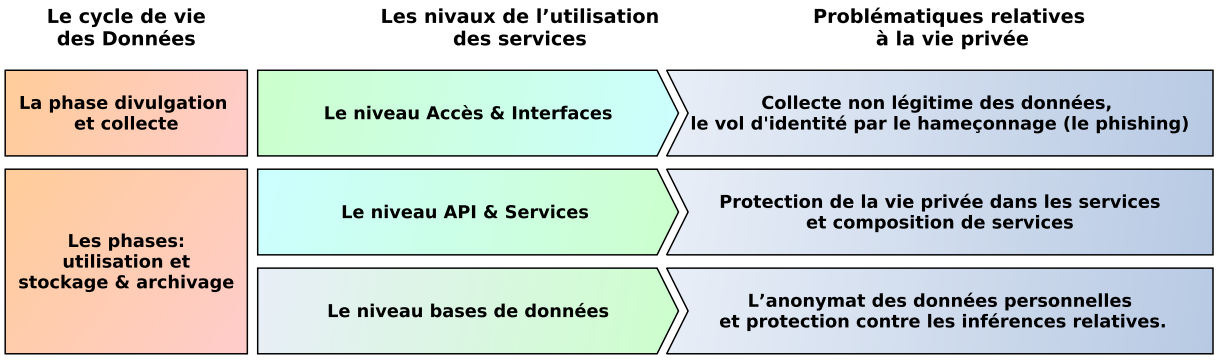


FIGURE 1.2 – Problématiques relatives à la vie privée des utilisateurs relatives au cycle de vie des Données et au niveaux d'utilisation des services.

Les menaces pour la phase de divulgation et collecte apparaissent lorsque l'utilisateur accède à un service en utilisant des interfaces de connexion (ex. page web, interface graphique d'une application mobile, ...). Les menaces dans cette phase sont multiples, les plus fréquentes sont : la collecte des données à l'insu de l'utilisateur (ex. adresse IP, localisation, ...), et le vol d'identité par le biais de l'hameçonnage (le phishing). pour la phase de l'utilisation, les menaces se manifestent lors de l'utilisation réelle d'un service. Le grand problème qui se pose dans cette phase c'est comment assurer que les données divulguées par l'utilisateur sont utilisées à la manière que ce dernier le souhaite (en termesde partage de diffusion et de transformation). Le problème devient plus complexe si le fournisseur de service fait

appelle à d'autres services pour satisfaire les besoins de l'utilisateur. Pour la phase de stockage & d'archivage, les menace apparaissent lorsque qu'il y a un besoin de partage ou de diffusion des bases de données qui contient les informations collectées. Dans ce cas, le risque de divulgation d'informations sensibles des utilisateurs provient. De ce fait, il est nécessaire de trouver des moyens efficaces pour assurer l'anonymat des propriétaires des données.

1.2 Problématiques de la thèse

Le but de notre thèse est de faire une étude sur la protection de la vie privée sur internet. Nous avons choisi de traiter cette problématique sur plusieurs niveaux, touchant ainsi les différentes phases de cycle de vie des données circulant sur Internet. Notons que c'est évident qu'on ne peut pas traiter toutes les problématiques relatives à la vie privée. De ce fait, nous avons traité une seul problématique pour chaque niveau d'utilisation de services (voir Figure 1.2). Ainsi, nous avons adopté un plan de travail qui traite les trois problématiques suivantes :

Problématique 1 (le niveau accès et présentation) : La lute contre le hameçonnage (Le Phishing)

Le Phishing est défini comme étant "l'acte frauduleux d'acquérir des données privées et sensibles, en se faisant passer pour une entité digne de confiance. Généralement, les données cibles sont de types : identifiants de connexion, mots de passe, les numéros de cartes de crédit ou de sécurité sociale, $\cdots$" [CTRL18]. Le choix de cette problématique est justifié par plusieurs raisons : i) Ce type d'attaque (le phishing) est le plus utilisé sur Internet pour le vol d'identité, ii) Malgré les outils existants, le nombre de victimes ne cesse de croître, iii) Les effets directs et affreux que causent le phishing sur les victimes.

Problématique 2 (le niveau API et services) : La protection de la vie privée dans les Web services

Au cours des dernières années, la problématique de la vie privée dans les services Web attire de plus en plus l'attention de la communauté de l'industrie et de la recherche. Un service Web a généralement sa propre politique de confidentialité qui définit un ensemble de règles applicables à tous les utilisateurs. La vie privée de services spécifie généralement trois types de politique : la politique d'utilisation, la politique de stockage, et la politique d'information. Dans le cadre de notre travail, nous affrontons le problème de la confidentialité dans les compositions des services Web. Cette problématique soulève plusieurs questions, parmi lesquelles :

Comment trouver une composition qui vérifie toutes les contraintes de confidentialité ? Comment représenter les politiques de confidentialité de manière à faciliter la vérification de l'absence de conflits ? Si la composition n'existe pas, est-ce que c'est possible d'établir un protocole de négociation pour éliminer les conflits et garder un maximum de protection ?

Problématique 3 (le niveau stockage et bases de données) : La protection des données personnelles

L'utilisation et le partage des données de type micro-données (micro-data) est devenu une nécessité dans plusieurs domaines telles que la santé, l'administration, l'économie ainsi que la recherche et l'enseignement universitaire. Le traitement et l'analyse de ces données peut entraîner des risques sur la vie privée et la confidentialité des individus. Le fait de supprimer les attributs identificateurs (nom, prénom, numéro de sécurité sociale, ...), ne donne pas un niveau de protection approprié. Le problème qui se pose c'est comment modifier les micro-donnés de façon a protégé les individus et en même temps garder l'utilité de ces données.

1.3 Contributions de la Thèse

Les principales contributions de cette thèse sont catégorisées en fonction des problématiques traitées dans cette thèse, nous les résumons comme suit :

1. Dans **la problématique de lute contre le Phishing**, nous avons proposé une approche de détection automatique des pages de phishing [Belabed12]. L'approche combine deux méthodes : une liste blanche personnalisée et une méthode à base de Machine Learning. La liste blanche est utilisée comme un filtre pour bloquer les pages imitant les pages légitimes usuelles d'un utilisateur. Les pages de phishing qui ne sont pas bloquées au niveau de la liste blanche sont traitées par un classificateur SVM dont nous avons proposé quelques attributs de classification. Les tests effectués sur le système proposé ont montré une amélioration par rapport à d'autres solutions.
2. Dans **la problématique de la protection de la vie privée dans les services Web**, nous avons proposé une approche de sélection qui préserve les exigences de vie privée des utilisateurs et les services Web, tout en minimisant le risque induit par l'utilisation des données privées par ces services [Belabed17]. Les contributions dans cette partie sont multiples, nous les résumons comme suit :
   — Premièrement, nous avons proposé deux modèles de protection de vie privée : un modèle de composition privée et un modèle de politique de

vie privée. Le modèle de composition est utilisé pour représenter le plan d'échange de données entre les services d'une composition. Le modèle de politique de vie privée est utilisé pour exprimer les préférences et les exigences en termes de vie privée des utilisateurs et des services. Le grand avantage de ce modèle est qu'il est compatible avec la norme P3P du framework de confidentialité w3c [Cranor02a, Ghazinour11].

— La deuxième contribution consiste à définir une fonction de divulgation de données à base d'intégrale floue [Grabisch96]. Cette fonction mesure le risque de menace à la vie privée causée par les fournisseurs de services lors de l'utilisation des données privées. Dans notre modèle de confidentialité, cette fonction est utilisée pour classer les compositions qui satisfont les exigences de vie privée, ce qui nous permet de sélectionner la composition qui porte le risque de menace minimal.

— Notre troisième contribution c'est la proposition de trois algorithmes de sélection. Le premier algorithme est une adaptation d'un algorithme de type meilleur d'abord que nous appelons CBFS (Constrained Best First Search). L'adaptation est faite pour gérer les contraintes de vie privée. Les autres algorithmes sont basés sur deux des modèles déclaratifs les plus populaires : la Satisfaisabilité booléenne (SAT)[Biere09] et le Answer Set Programming (ASP)[Janhunen16]. L'efficacité de ces algorithmes est testée et comparée sur différents types de données.

3. Dans **la problématique de la protection des données personnelles**, nous avons proposé une approche de protection qui se base sur la publication des données fictives au lieu de vrais données [Belabed14]. Les données fictives sont générées en utilisant des modèles issues des données originales en se basant sur les techniques de Machine Learning. Les données générées gardent certaines propriétés des données originales, ce qui assure à la fois une forte protection et une grande utilité.

1.4 Organisation de la Thèse

En plus de ce chapitre d'introduction, ce document est composé de quatre grands chapitres, suivis d'une conclusion générale où nous résumons nos principaux contributions ainsi qu'un aperçu sur les perspectives de ce travail.

Le deuxième chapitre présente un état de l'art sur la vie privée sur Internet. Nous détaillons dans ce chapitre les principaux concepts relatifs à la vie privée, ses différentes facettes, les types d'attaques relatives ainsi que les différents niveaux de protection. Enfin, une synthèse des principaux technologies de protection de la vie privée est introduite.

Les trois chapitres suivants sont consacrés à chacune des problématiques traitées dans cette thèse.

Le troisième chapitre traite la problématique de protection des données personnelles contre le Phishing. La première partie de ce chapitre est consacrée à un état de l'art sur le Phishing, à savoir, les définitions, les différents types d'attaques, ainsi qu'un survol des solutions anti-phishing. La second partie introduit les détails de l'approche proposée. Cette partie présente aussi les résultats des tests ainsi que la discussion de ces résultats.

Dans le quatrième chapitre nous abordons la problématique de protection de la vie privée dans un contexte de sélection de services Web. Ce chapitre introduit dans un premier lieu un état de l'art sur les travaux réalisés dans ce domaine. Ensuite, une description détaillée du Framework de sélection ainsi que les algorithmes de sélection proposés est présentée. La fin du chapitre est marquée par les expérimentations et la discussion des résultats de l'évaluation.

Le cinquième chapitre traite la problématique de l'anonymat des données personnelles. Après l'introduction des concepts de bases relatives aux données à caractère personnelles, nous mettons l'accent sur les différentes approches proposées pour assurer l'anonymat de ces données. L'approche proposée est ensuite introduite avec les détails de l'implémentation ainsi que les résultats de l'évaluation.

Privacy isn't about hiding something. It's about being able to control how we present ourselves to the world.

Bruce Schneier, Cryptography and Security Specialist

2

La vie privée sur Internet : Etat de l'art

▷ *L'objectif de ce chapitre est de donner une vue générale sur certains aspects de de la vié privée, pour mieux assimiler les différentes problématiques traitées dans cette thèse.* ◁

Plan du chapitre

2.1 Introduction

Le concept de "vie privée" (privacy en anglais) est un concept général, qui a un sens différent selon les personnes et varie selon chaque individu, et dépend aussi du contexte [Moore13]. Ce concept peux englober plusieurs notions telles-que : le droit d'être seul, le droit à la liberté de pensée, le droit à une propre vie familiale, le droit de protéger sa réputation, etc. De plus, ces notions peuvent varier d'un contexte à un autre [Mendel13]. Sur Internet, la vie privée englobe toute une gamme de questions touchant à la capacité des utilisateurs d'Internet à assurer le contrôle, et la transparence sur l'utilisation de leurs données personnelles lorsqu'elles sont recueillies et utilisées par des entités privées ou publiques. L'objectif de ce chapitre n'est pas d'introduire une étude détaillée sur la vie privée, mais de donner une vue générale sur certains aspects de cet important concept, pour mieux assimiler les différentes problématiques traitées dans cette thèse.

2.2 Définitions

Comme nous l'avons déjà mentionné, le concept de vie privée est un concept assez général et assez flou et difficile à cerner toutes ces facettes. Cela à répercuter sur les multiples définitions dans la littérature.

L'une des premières définitions a été donnée par A. Westin [Westin68, Westin03]. Cette définition a associé le terme de vie privée (ou protection de la vie privée) à la revendication d'un individu, d'un groupe ou d'une institution, à pouvoir contrôler les conditions d'acquisition et d'utilisation de leurs informations personnelles. Il s'agit également de savoir quand ces informations seront obtenues et quelles utilisations en seront faites par d'autres. Bien que cette définition a été introduite au début pour un monde hors-ligne, elle est aujourd'hui largement adoptée et étendue par plusieurs académiques pour définir la vie privée dans le monde numérique [Ginosar17, Schwaig06, Oulasvirta14, Gerlach15].

D'autres chercheurs [Solove02, Moore13, Moore10] ont classifié les le concept de vie privée en six catégories : (1) le droit d'être laissé seul ; (2) le secret ; (3) l'intimité ; (4) Avoir le contrôle sur l'information ; (5) l'accès restreint au soi ; et (6) un regroupement de plusieurs catégories.

La définition de la vie privée comme étant «le droit d'être laissé seul (ou tranquille)» a été donnée la première fois en 1890 par Warren & Brandeis [Warren90] comme réponse à l'utilisation intense de la photographie. Cette définition a été largement critiquée comme étant trop vague [Moore13]. En plus, ce n'est pas chaque violation du droit d'être laissé seul est une violation de la vie privée. L'exemple le plus simple c'est le frôlement de deux personnes sur un trottoir occupé.

L'une des définition de la vie privée qui tourne au tour du secret, a été donnée par Richard Posner [Posner81], il a défini la vie privée comme : «le droit de dissimuler des faits discréditables sur soi-même ». Cette définition a été critiquée [DeCew97], du fait que les informations secrètes ne sont pas toujours privées (par exemple, des plans militaires secrets) et aussi, les affaires privées ne sont pas toujours secrètes (par exemple, les dettes d'une personne).

Une opinion très répandue qui dit que la vie privée et l'intimité sont étroitement liées. Dans ce cadre, Julie Inness [Inness96] a défini la vie privée comme : «*l'état de l'agent ayant le contrôle sur les décisions concernant des sujets qui tirent leur signification et leur valeur de l'amour, la compassion, ou les goûts,de cet agent*». Cette définition a été critiquée [Solove02], du fait qu'il est possible d'avoir des relations privées sans elles soient intimes et aussi d'accomplir des actes privés qui ne sont pas intimes.

Avoir le contrôle sur les informations personnelles a également été proposé comme définition de la vie privée. Dans ce contexte, Alan Westin [Westin68], a donné la définition suivante : «*La vie privée n'est pas simplement une absence d'information sur nous dans l'esprit des autres ; c'est plutôt le contrôle que nous avons sur ces informations*». Les critiques sur cette définition se basent principalement sur le fait que cette définition ne couvre pas l'aspect physique de la vie privée [Moore13], comme le contrôle de l'accès aux lieux et aux corps.

Parmi les définitions de la vie privée comme «l'accès limité au soi», celle donnée par S. Bok [Bok89]. Il a défini la vie privée comme : «*la condition d'être protégé contre tout accès non désiré par d'autres personnes*». Malgré que cette définition couvre aussi bien l'aspect numérique que physique, elle a été critiquée d'être trop étroites [Solove02].

Enfin, beaucoup considèrent la vie privée en tant que concept de cluster qui comprend plusieurs des dimensions mentionnées ci-dessus. Par exemple J. DeCew [DeCew06], a proposé que la vie privée est un concept qui s'étend sur l'information, l'accès et les expressions. Moore [Moore10] a argué que la vie privée est le droit de contrôler l'accès et l'utilisation des informations personnels et les localisations spatiales.

2.3 Vie privée et Législation

L'importance de la législation relatives à la protection de la vie privée réside dans le fait qu'elle constitue l'élément clé pour le développement du commerce et de la collaboration électronique à travers le monde. La figure 2.1 montre l'existence ou l'absence des lois de protection de données personnelles dans les pays du monde, ainsi que la compatibilité de ces lois avec celles de l'union Européenne.

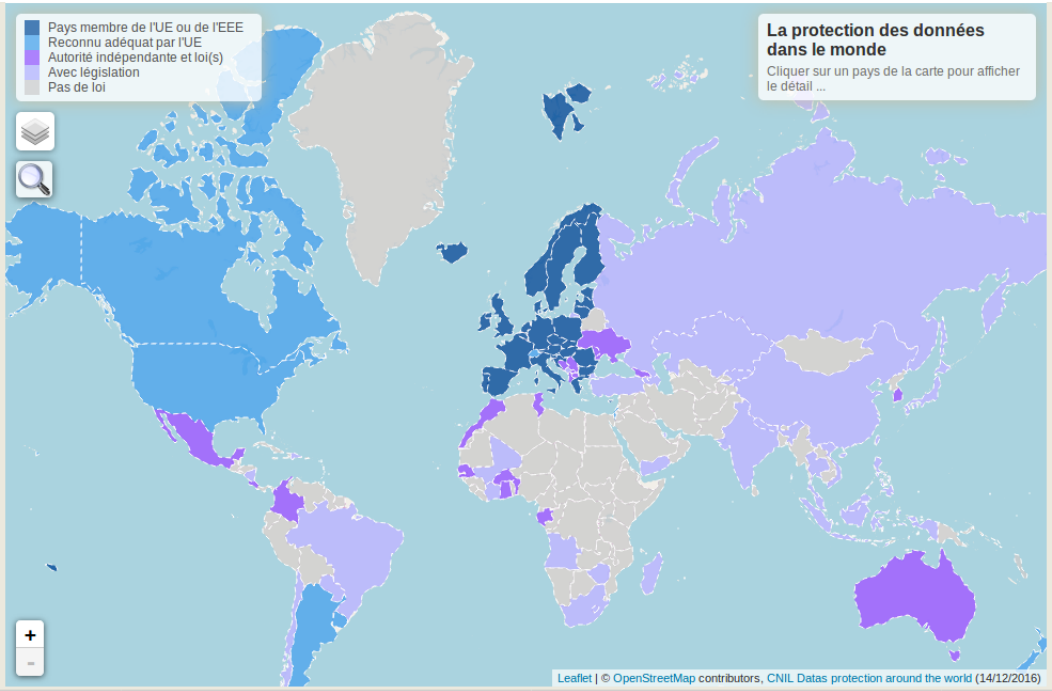


FIGURE 2.1 – La protection des données dans le monde (source[cni16])

Cette section survole les principales lois relatives à la protection de la vie privée à travers le monde. Nous nous intéressons en particulier par : La loi islamique, la loi algérienne, les lois de l'union européenne, la loi internationale évoqué par les Principes directeurs de l'OCDE et enfin les Initiatives HIPAA des États-Unis.

2.3.1 Vie privée Dans l'islam (Shariah)

L'Islam accorde une grande importance au droit humain à la vie privée. Cette importance, couvre différents aspects de la vie privée : l'aspect physique, l'accès et l'utilisation des informations personnelles, l'intimité et réputation. Cela est répercuté dans plusieurs versets du Coran et du Hadith. [Hayat07].
Dans l'aspect physique de la vie privée, l'islam a interdit tout accès non autorisé à des propriétés privées, dans ce cadre, Allah a dit :"*ô vous qui croyez ! N'entrez pas dans des maisons autres que les vôtres avant de demander la permission et de saluer leurs habitants. Cela est meilleur pour vous. Peut-être vous souvenez-vous*" [1].
Le Prophète (paix et bénédiction de Dieu sur lui) est allé jusqu' à dire qu'un homme ne devrait pas entrer, même dans sa propre maison soudainement ou discrètement [2].
l'interdiction d'acquérir ou de divulguer des informations personnelles sans autorisation est claire dans le verset coranique suivant :"*ô vous qui avez cru ! évitez de trop conjecturer [sur autrui] car une partie des conjectures est péché. Et n'espionnez*

1. Sourate Al-Nur, verset 27.
2. Sahih Muslim, Hadith N° 3555.

pas ; et ne médisez pas les uns des autres. L'un de vous aimerait-il manger la chair de son frère mort ? (Non !) vous en aurez horreur. Et craignez Allah. Car Allah est Grand Accueillant au repentir, Très Miséricordieux"[3].
Il y a aussi des peines en cas d'atteinte à la réputation des personnes (et spécialement les femmes), dans ce contexte, Allah a dit : "*Et ceux qui lancent des accusations contre des femmes chastes sans produire par la suite quatre témoins, fouettez-les de quatre-vingts coups de fouet, et n'acceptez plus jamais leur témoignage. Et ceux-là sont les pervers.* "[4]

2.3.2 La loi algérienne

Comme la figure 2.1 le montre, et jusqu'à l'écriture de ces lignes[5], l'Algérie ne dispose pas d'une loi en vigueur pour la protection de la vie privée et les données à caractère personnel. Le 27 décembre 2017, le conseil des ministres a adopté un projet de loi relatif à la protection des personnes physiques dans le traitement des données à caractère personnel [Service17]. Selon le communiqué à l'issue de la réunion du conseil des ministres [Service17], ce nouveau texte "accompagnera le développement du traitement numérique des données administratives, juridiques et financières, dans des secteurs de plus en plus nombreux du service public" et "régulera la protection des personnes physiques lors du traitement de leurs données à caractère personnel". le projet de loi énonce notamment "l'exclusion des données à usage privé exclusif du traitement en l'objet, la nécessité de l'accord de la personne concernée lors du traitement de ses données personnelles, sauf dans des situations d'obligations légales, essentiellement judiciaires, l'institution d'une protection renforcée pour la protection des données personnelles de l'enfant et l'institution d'une Autorité nationale de protection des données à caractère personnel, placée auprès du Président de la République" [Service17].

2.3.3 Directives de l'Union Européenne.

2.3.3.1 La directive européenne 95/46/CE

La directive européenne 95/46/CE « relative à la protection des personnes physiques à l'égard du traitement des données à caractère personnel et à la libre circulation de ces données » [Directive95]. Le but de cette directive est d'établir un ensemble de règles communes relatives au traitement des données personnelles. Ces règles doivent être mis en oeuvre par chaque état membre afin de protéger la vie privée et assurer la libre circulation des données personnelles en Europe. Cette

3. Sourate Al-Hujurat, verset 12.
4. Sourate Al-Nur, verset 4.
5. Le 15/01/2018

directive repose sur les recommandations initialement établies par l'OCDE, et sept principes ont été repris : La limitation de la collecte des données ; la qualité des données collectées ; la participation des propriétaires des données ; la limitation de l'utilisation des données ; les mesures de sécurité ; la spécification du but de l'utilisation des données et la responsabilité judiciaire [Directive95, Léonard99].

2.3.3.2 Règlement général sur la protection des données (RGPD) n° 2016/679

Ce réglement remplace la directive européenne 95/46/CE. Le réglement est adopté par le parlement européen et le conseil européen en avril 2016 et deviendra applicable en mai 2018 [dir08]. Le nouveau règlement s'étend aux exigences précédentes de collecte, de stockage et de partage des données personnelles et nécessite que le consentement du sujet soit donné explicitement. Le règlement impose aux organisations la notion de « Privacy by design», c-a-dire, de prendre en compte des exigences relatives à la vie privée dès la conception des produits et des systèmes informatique. Il impose aussi la nomination obligatoire d'un délégué à la protection des données (Privacy officer), dont le rôle principale est de contrôler le respect du règlement [Mittal17].

2.3.4 Les Principes directeurs de l'OCDE sur la protection de la vie privée

L'Organisation de Coopération et de Développement Économiques (OCDE) est un forum pour les pays engagés dans "la démocratie et l'économie de marché". L'Organisation fournit un cadre dans lequel les gouvernements comparent les expériences politiques, cherchent des réponses à des problèmes communs, identifient les bonnes pratiques et coordonnent les politiques nationales et internationales [oec18b]. Les Principes directeurs de l'OCDE sur la protection de la vie privée, ont été développées à la fin des années 1970 et adoptées en 1980. Ils sont devenus un ensemble de règles internationalement acceptées pour le traitement des informations personnelles. Le principale objectif est de protéger les données à caractère personnel, tout en poursuivant la libre circulation de l'information entre les différentes organisations, éventuellement entre les pays [oec18a]. Les directives de l'OCDE définissent huit principes : Limitation de la collecte des données, la qualité des données collectées, la participation des propriétaires des données, la spécification du but de l'utilisation des données, limitation de l'utilisation des données, Les mesures de sécurité, la transparence et la responsabilité judiciaire. Plus de détails sur ces principes peuvent être trouvés dans [oec18b].

2.3.5 Initiatives HIPAA (États-Unis)

HIPAA (the Health Insurance Portability and Accountability Act) [Gunn04], la loi sur la transférabilité et la responsabilité de l'assurance-maladie. La loi a été adoptée par le Congrès américain en 1996. Le but principale de cette loi est de rendre la prestation des soins de santé plus efficace et à accroître le nombre d'Américains ayant une couverture d'assurance-maladie. Une partie de cette loi traite la protection des données médicales. Le règlement relatif à la vie privée dans cette loi, exige que les organisations et les professionnels de la santé, ainsi que leurs associés, élaborent et suivent des procédures garantissant la vie privée et la sécurité des renseignements médicaux lorsque ces derniers sont transférés, reçus, manipulés ou partagés. Cela s'applique à toutes les formes de données (en format papier, oral ou électronique).

2.4 Principes de la vie privée

Pour assurer une protection de la vie privée, il y a des principes fondamentaux à respecter. Ces principes doivent être implémentés dans tout système ou architecture compatible avec la protection de la vie privée. Notons que la majorité de ces principes sont inspirés des différentes législations susmentionnées. Parmi ces principes nous citons [oec18b, dir08] :

- **La Minimisation des données** : Ce principe exige que les données devant être collectées pour des finalités déterminées, légitimes et limitées à ce qui est nécessaire au regard des finalités pour lesquelles elles sont collectées.
- **Le Consentement explicite** : Ce principe exige que toute collecte ou traitement de données personnelles ne doit avoir lieu sans l'autorisation explicite de la part du propriétaire de ces données.
- **La Souveraineté des données** : Ce principe donne à un individus le droit d'accès à ses informations personnelles, le droit de les corriger si elles sont inexactes, incomplètes ou périmées et éventuellement les supprimer (Droit à l'oubli).
- **La Transparence** : Ce principe impose qu'un utilisateur à le droit d'être informé sur ces informations personnelles collectées, et comment et pourquoi ses données sont utilisées et avec qui elles sont partagées.
- **La Sécurité** : Une agence qui détient des renseignements personnels doit assurer des mesures de sécurité contre la perte, l'accès non autorisé, l'utilisation abusive ou la divulgation.

D'autre principes sont relatifs à l'identité et les actions d'un individu sur Internet, parmi ces principes nous citons [Pfitzmann10] :

— **L'Anonymat** : Ce principe permet à un utilisateur de réaliser une action sans que cela puisse être reliée à son identité.
— **La Pseudonymat** : Ce principe définit le fait qu'un système offre à ses usagers la possibilité d'agir sous un pseudonyme au-lieu de leurs vrais identités.
— **La Non-chaînabilité** : Ce principe définit le fait que dans un système, un attaquant soit incapable de relier deux actions anonymes qui ont été menés par le même individu.
— **La Non-observabilité** : Ce principe définit le fait qu'un attaquant soit incapable de savoir, à un instant donné, si une action particulière a lieu ou non.

2.5 Les Menaces sur la Vie privée

Toutes les menaces relatives à la vie privée tournent au tour de l'utilisation non autorisée et/ou malveillante des données collectées (d'une manière légale ou illégale). Cette section donne un aperçu sur les menaces les plus fréquentes.

2.5.1 Divulgation des données personnelles : Atteinte à la réputation et à l'intimité

Avec l'émergence des réseaux sociaux, les services de partages des photos et des vidéos, trouver et accéder à des informations personnelles est devenu une opération très simple. Des informations comme la date de naissance, l'emploi, la situation familiale, les préférences musicales, les informations comportementales, etc. qui sont considérées par la majorité d'entre nous comme triviales et inoffensives peuvent être utilisées contre nous. En accédant a ces informations, les risques de préjudice, d'inégalité, de discrimination et de perte d'autonomie apparaissent facilement, les exemples dans ce cadre ne manquent pas [Solove06]. Par exemple, nos amis et nos proches ont maintenant moins de difficulté à savoir où nous sommes, ce qui cause dans certains contextes des situations gênantes. Aussi les employeurs ont tendance à utiliser des informations en ligne pour éviter d'embaucher des personnes qui correspondent à certains profils. Si les informations divulguées sont sensibles, on peut faire face à des situation plus délicates, qui touche directement la dignité et réputation, et arrive jusqu'au l'harcèlement et le chantage.

2.5.2 Vol et usurpation d'identité

Le vol d'identité [Whiting13] est l'un des crimes qui connaît la plus forte croissance. On peut considérer qu'il s'agit d'un vol d'identité, chaque fois qu'un criminel

s'empare d'une partie des données d'une personne et les utilise à son propre profit. Étant donné que de nombreux organismes privés et gouvernements conservent des informations sur les individus dans des bases de données accessibles, les voleurs ont une occasion inépuisable de les récupérer et de les utiliser à mauvais escient. En général, le vol d'identité comporte le vol du numéro de sécurité sociale, du numéro de carte de crédit ou d'autres informations personnelles dans le but d'emprunter de l'argent, de faire des achats et d'accumuler des dettes. Dans certains cas, les voleurs retirent même de l'argent directement du compte bancaire de la victime [ukg17]. Au-delà des pertes financières, il peut y avoir d'autres conséquences importantes et graves pour les victimes. Par exemple, si une personne utilise l'identité d'une autre personne pour commettre un crime. La victime peut se retrouver dans le cadre d'une enquête criminelle, et dans la difficulté de prouver son innocence. Et cela sans compter les effets désastre que cette incidence peut causer sur son état psychologique et mental et ces relations professionnelles et sociales.

2.5.3 Le Profilage

Le profilage désigne le fait de compiler des dossiers d'information sur des individus afin de déduire des intérêts et des caractéristiques par corrélation avec d'autres profils et données [Ziegeldorf14]. Les informations utilisées dans le profilage contiennent : des données identificateurs tels-que : les adresses IP, les numéros d'identification des navigateurs web et les systèmes d'exploitations, etc. Et des données concernant les activités et le comportement des utilisateurs sur Internet, comme : les requêtes sur les moteurs de recherche, les sites visités, les relations et les communications sur les réseaux sociaux, les produits achetés sur Internet, etc. Le profilage n'est pas une menace en soi, il y a des grandes avantages de l'utilisation de cette technique dans plusieurs domaines. Les systèmes de recommandation [Aggarwal16] qui proposent aux clients des produits et des services qui correspondent à leurs préférences et leurs intérêts, est un bon exemple de l'utilisation bénéfique de cette technique. Le profilage devient une menace sur la vie privée dans deux cas : premièrement, si les données utilisées dans le profilage sont collectées d'une manière illégale en utilisant les diverses techniques de tracking et de surveillance [Bujlow17]. Deuxièmement, si les profils issus après le traitement des données collectées, sont utilisés pour des mauvaises fins, comme : la discrimination par les prix [Odlyzko03], les publicités non sollicitées et nuisibles [Chen15], les spams [Bhowmick18], etc.

2.6 Les techniques d'attaques

Quelque soit le type de menace sur la vie privée, cette dernière commence soit par une collecte ou un accès non autorisé à des données personnelles, ou-bien des inférences faites à partir de la combinaison des données publiques provenant de plusieurs sources (eg. Liste des électeurs). Cette section présente les techniques d'attaques les plus utilisées. Nous introduisons, en particulier, les techniques à base de : Malwares, Cookies, Phishing, et attaques par inférences.

2.6.1 Les Malwares

Un Malware (malicious software) [Bai16], Désigne généralement, une famille de programmes conçus à des fins malveillantes. Ces fins peuvent être : exécuter des programmes intrusifs, détruire des données, voler des informations sensibles et compromettre la crédibilité et la disponibilité de l'ordinateur-smartphone-tablette de la victime. Cette famille de programme peut inclure : les virus, les vers, les chevaux de Troie, les Spywares, les Bots, les Rootkits, les Ransomware, etc. Tous ces types de programmes sont largement utilisés pour l'accès non autorisé, le vol des données personnelles et sensibles, ainsi que les publicités non sollicitées et nuisibles. À titre d'exemple, en 2017, le Ransomeware «WannaCry» [jun17] a pu infecter plus de 300 000 machines dans 150 pays. Dans la même année, le Adware «CopyCat» [Ltd17] a infecté 14 millions appareils Android, toute en faisant un bénéfice de plus 1,5 million de dollars en faux revenus publicitaires aux hackers derrière l'opèration. Notons que malgré les solutions anti-Malwares fournies par les grandes firmes de sécurité informatique (Kaspersky Lab, Symantec, McAfee, etc.), la détection et l'élimination des malwares reste toujours une problématique ouverte et un domaine actif de recherche [Ahvanooey17, Hong18, Bai16, Yan17, Ye17].

2.6.2 Les Cookies

Les cookies [Barth11] sont des petits fichiers texte qui résident sur les appareils des utilisateurs du Web. Les informations qu'ils contiennent sont définies et accessibles par les serveurs les sites Web visités. Dans leur principe de conception, les cookies sont conçus pour améliorer et faciliter l'expérience de navigation des internautes. À travers les cookies, les serveurs peuvent se souvenir et identifier certaines informations concernant les utilisateurs. Ces informations peuvent être : la date de visite, les items consultés et achetés, les préférences de configuration comme la langue et l'affichage, les scores de jeu, etc. mais aussi des informations sensibles telles que les mots de passe et les numéros de cartes de crédit. Le problème est que les cookies peuvent également être utilisés pour le tracking, l'enregistrement et

l'analyse du comportement des utilisateurs. D'ailleurs, les cookies sont considérés comme étant la méthode de traking la plus répandu[moz17]. Le problème provient particulièrement de ce qui est appelé les « *cookies tiers* » [Parsons15]. L'origine de ces derniers sont les composants tiers qui se trouvent dans la majorité des site Web populaires. Ces composants peuvent être : des images, des pixels, des bannières publicitaires, ou bien d'autres formes de composants comme les boutons Facebook et Twitter. À chaque fois qu'un utilisateur visite une page Web contenant des composants tiers, les organismes propriétaires de ces derniers, et à travers les cookies, peuvent faire le lien entre l'utilisateur et la page visité. La collaboration entre les

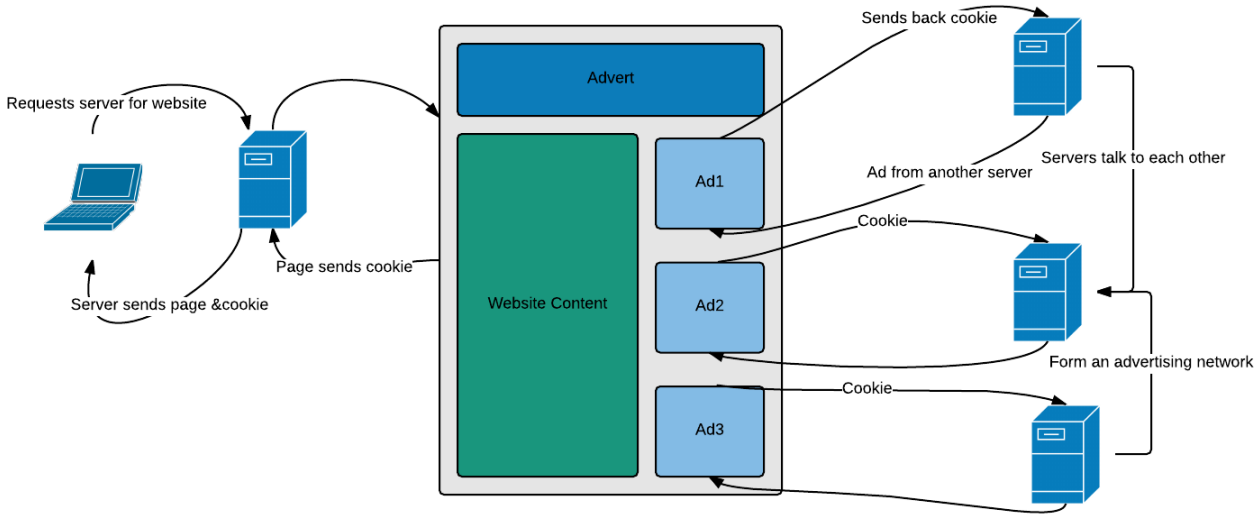


FIGURE 2.2 – Principe des réseaux publicitaires.

organismes tiers dans le cadre des réseaux publicitaires (La figure 2.2 présente le principe) permet de construire un profil assez complet sur les préférences et les habitudes des utilisateurs du Web. Un autre problème provient des « *cookies de session* » [Parsons15]. Ce type de cookies permet à un site Web de garder une trace des mouvements des utilisateurs de page en page, l'objectif est de faire éviter les utilisateurs de fournir chaque fois les mêmes informations déjà fournies sur le site. Des informations comme : les items mis au panier, les achats validés et surtout les détails de l'authentification. En utilisant des techniques à base d'injection de code, comme le Cross-site scripting (XSS)[Jakob13], un attaquant peut voler un cookie de session. Cela lui permet de se connecter à un site Web protégé par un nom d'utilisateur et un mot de passe et exécuter par la suite des actions malveillantes [Gupta17, Vogt07].

2.6.3 Le Phishing

L'hameçonnage (Le phishing) [Group17] est considéré comme l'une des attaques les plus répandus, entraînant des pertes financières importantes pour les entre-

prises et les utilisateurs. À titre d'exemple, le site PhishTank[6] a reporté plus de 12000 attaques, et cela seulement pour le mois de Mars 2017. La méthode la plus utilisée pour réaliser ce type d'attaque est à travers des e-mails frauduleux. Ces derniers, redirigent les victimes vers des sites et des liens forgés pour ressembler au maximum aux sites cibles au phishing [wom18]. La figure 2.3 représente un exemple d'une page de phishing qui imite la page d'accueil de la plateforme de stockage Dropbox[7].

Les techniques de phishing ainsi que les principales solutions de lute contre le

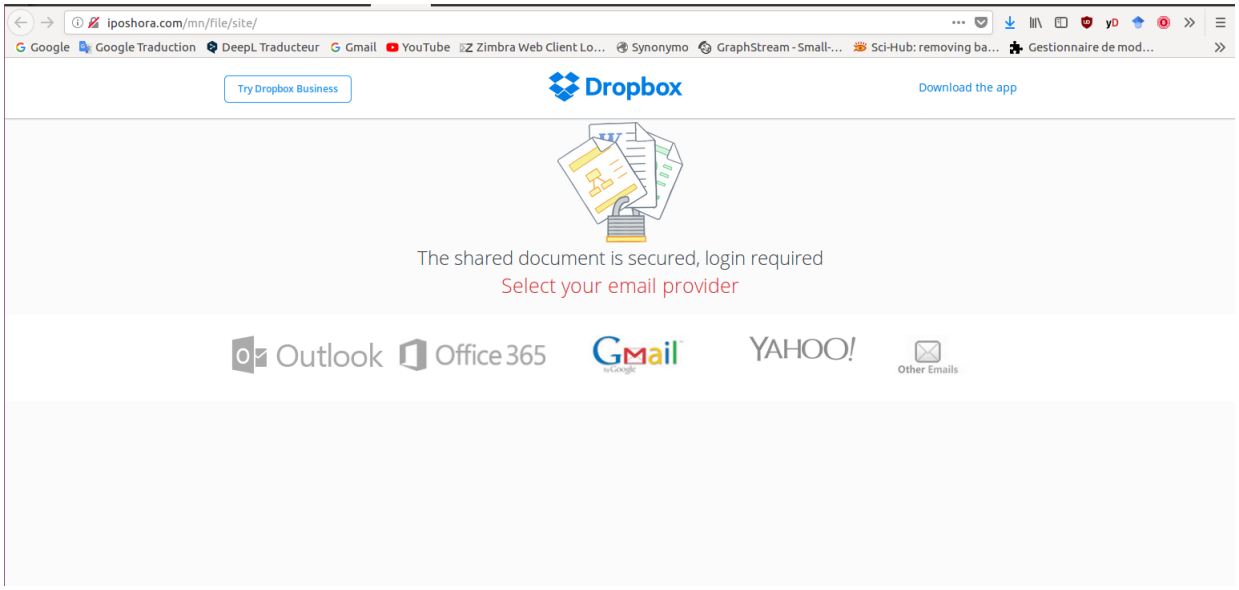


FIGURE 2.3 – Exemple d'une page de phishing ciblant la plateforme Dropbox.

phishing seront présentées en détail dans le chapitre 3.

2.6.4 Attaques par Inférence

On désigne par « attaque par inférence » toute tentative d'inférer des nouvelles informations personnelles en combinat : des données public et non sensibles comme les listes des électeurs, des données anonymisées (eg. par simple suppression des données nominatives : nom, prénom, NSS, etc. ou bien avec techniques plus avancées) ainsi que des connaissances auxiliaires. Les techniques d'inférences sont diverses et variées, les principales catégories de ces techniques sont : les méthodes statistiques et probabilistes, les méthodes à base de Data-mining et d'intelligence artificielle et les techniques de chaînages qui consistent à relier les enregistrements de deux ou plusieurs bases de données différentes contenant un ensemble d'individus en commun [Chen09, Wang10]. Les attaques par inférences peuvent cibler les différentes types de données : les micro-données (tables

6. https://www.phishtank.com/index.php
7. https://www.dropbox.com/

contenant des informations structurées sur les individus comme l'âge, l'adresse, le niveau d'études, le statut professionnel, etc.), les données transactionnelles (ex. les enregistrements des clients et leurs achats) ainsi que les données spatio-temporelles (données de géolocalisation des individus et leurs déplacements). Les inférences faites sur ces types de données et qui menacent la vie privée des individus sont énormes. À titre d'exemples, pour les micro-données, les informations inférées sont souvent les maladies et le revenu des individus [Wang10]. Pour les données transactionnelles, établir un lien entre un client et ses achats, peut révéler beaucoup d'informations sur ses préférences, ses maladies et ses habitudes de consommation [Li14, Sui17]. Les inférences sur les données spatio-temporelles sont aussi importantes, un attaquant peut identifier les points d'intérêts caractérisant un individu comme sa maison, son lieu de travail, son restaurant préféré, etc. il peut aussi prédire sa prochaine localisation et même découvrir ses relations sociales [Gambs10, Gambs12, Gambs14]. Notons que dans notre thèse nous nous intéressons plus aux micro-données. Un état de l'art sur les différents types d'attaques et les approches de protection relatives sera présenté au chapitre 5.

2.7 Les Technologies de Protection de la Vie privée

Dans la présente section, nous allons présenter les principales technologies et approches de protection de la vie privée. Nous introduisant en particulier : les systèmes de gestion d'identité, les systèmes de communication anonyme, les accréditations anonymes, ainsi que quelques standards relatifs à la protection à base de langages de spécification des exigences de vie privée . Nous omettons dans cette section, les approches anti-phishing et les approches d'anonymisation des données. Ces dernières seront introduites respectivement aux chapitres 3 et 5.

2.7.1 Les systèmes de gestion d'identité

Les systèmes de gestion d'identité sont définis comme étant des systèmes ou des Frameworks qui administrent la collecte, l'authentification ou l'utilisation des identités et les informations liées à ces identités [Hansen08]. Toutes ces opérations impliquent la divulgation de beaucoup d'informations personnelles pour assurer l'authenticité et la vérification de l'identité des utilisateurs de système. Cela augmente le risque des menaces sur la vie privée des utilisateurs en élargissant les possibilités de profilage, de suivi des activités des utilisateurs et le vole des informations. Le risque s'accroît encore si un individu utilise plusieurs systèmes avec différents entités de gestion d'identité, ce qui est souvent le cas. Cette situation conduit à un nombre croissant d'identités différentes que chaque utilisateur doit gérer. Par conséquent, beaucoup de gens se sentent surchargés d'identités et

souffrent de la fatigue des mots de passe [Jøsang07]. Pour renforcer la protection de la vie privée dans les IMS, plusieurs solutions ont été proposées. La majorité de ces solutions se basent sur le principe de Single-Sign-On [Ado16] . Ce principe repose sur le fait qu' un utilisateur connecte à un système (le gestionnaire d'identité) et se voit automatiquement accorder l'accès à d'autres services (voir figure 2.4). Plusieurs implémentations de ce mécanisme existent dont on cite : OpenID

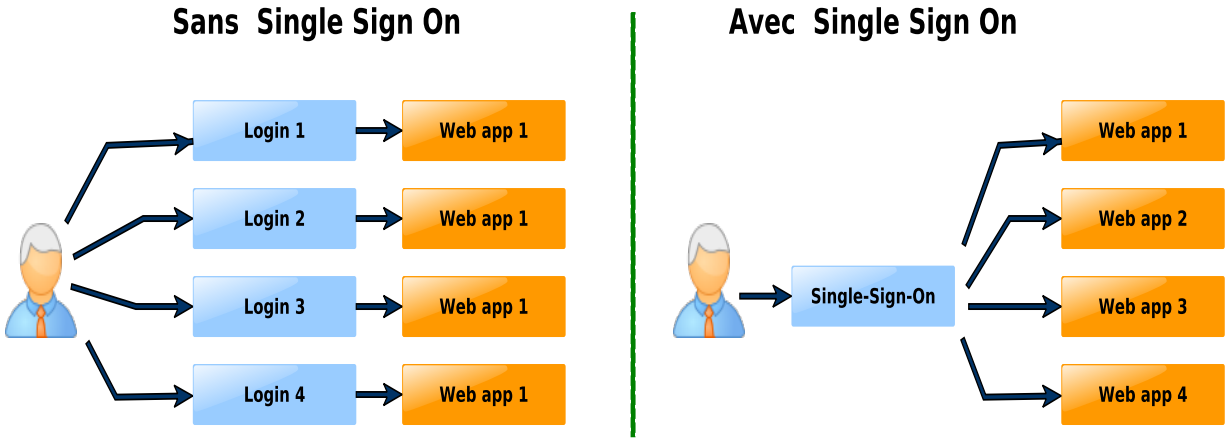


FIGURE 2.4 – Single-Sign-On vs Authentification classique.

[Recordon06], SAML [Simon04], Facebook Connect [Morin08], Microsoft Account [mic18], etc. Malgré que ce type de solution diminue la dissémination des informations personnelles sur plusieurs entités et évite la gestion d'un grand nombre de comptes en réduisant ainsi le phénomène de «password fatigue », un tel système est considéré par les Hackers comme un « Big Phish ». Si un Hacker compromet un compte Single-Sign-On, il obtient évidement un accès non autorisé à tous les systèmes liés. D'autres principes en faveur de la protection de la vie privée sont aussi implémentés dans les IMS. Des principes comme : l'anonymat (pseudonymat), la non-chaînabilité et la minimisation de collecte des attributs identificateurs. En général, ces principes sont implémentés à l'aide de plusieurs techniques, les plus importantes sont : l'identité virtuelle [Sarma08, Aguiar07] et les accréditations anonymes (voir la section 2.7.2). Le concept de l'identité virtuelle consiste à conserver une identité principale et de laisser le IMS gérer et mapper cette identité à des identités virtuelles en fonction du contexte et de service demandé. Ces identités virtuelles doivent être suffisamment anonymes pour qu'il soit difficile d'établir des liens entre eux. Les accréditations anonymes [Chaum85] sont un ensemble de techniques qui permettent à un utilisateur de prouver une propriété (eg. Age > 18, la nationalité) ou un droit d'accès à un service (eg. par l'appartenance à un groupe), mais sans révéler son identité ou des informations relatives à son identité. Plusieurs Systèmes implémentent ces principes, parmi lesquelles on cite : Privacy

and Identity Management for Europe (PRIME) [PRIME05], Future of Identity in the Information Society (FIDIS)[Rannenberg09], Liberty Alliance [Alliance02] et Identity Mixer (Idmix)[Camenisch02].

2.7.2 Accréditations anonymes

Les accréditations anonymes est un concept introduit par David Chaum [Chaum83, Chaum85], et réalisé la première fois par Camenisch et Lysyanskaya [Camenisch01]. C'est un ensemble de techniques à base cryptographique, permettant à un utilisateur de prouver qu'il possède une propriété ou un justificatif délivré par une organisation (une accréditation), sans révéler quoi que ce soit sur lui-même d'autre que la possession de l'accréditation. La forme la plus simple d'accréditations anonymes est lorsqu'un utilisateur veut prouver à un vérificateur (ex. fournisseur de service) qu'il possède un ou plusieurs attributs, comme par exemple : l'age> 18, l'adresses est à l'ouest algérien, etc. Dans ce cas, l'utilisateur doit fournir un certificat de la part d'un tiers de confiance (eg. Gouvernement), pour convaincre le vérificateur de l'exactitude des attributs demandés. La solution la plus simple est de convaincre le vérificateur qu'un prédicat complexe sur les attributs est valide [Neven08]. Un prédicat peut être une opération arithmétique (+,-,*,etc.), comparative(<, <, =, etc.) ou logique (OR, AND, etc.). Pour l'exemple précédent , le certificat délivré au vérificateur peut être simplement :

$$((anne\_naiss < cette\_anne - 18) \wedge ((adr = Tlemcen) \vee (adr = S.Belabbes) \vee (adr = Oran)))$$

Pour certifier des attributs plus complexes ou des messages, d'autres techniques plus compliquées sont utilisées, comme : la signature aveugle [Chaum83], la signature de groupe [Chaum91] et la preuve de connaissance à divulgation nulle.

2.7.2.1 Signature Aveugle

Concept introduit par David Chaum en 1983 [Chaum83]. Ce concept offre la possibilité de faire signer un message (signature numérique [Diffie76]) sans que le signataire puisse lire le contenu de ce message. Le principe de base pour réaliser ce type signature se base sur une fonction d'aveuglement (qui chiffre le contenu d'un message) et son inverse, la procédure se déroule comme suit (voir figure 2.5) :

1. le propriétaire de message "aveugle" le message msg (une fonction de chiffrement), avec un nombre aléatoire b (le facteur aveuglant), le résultat sera : $aveugle(msg, b) = msg_{av}$.

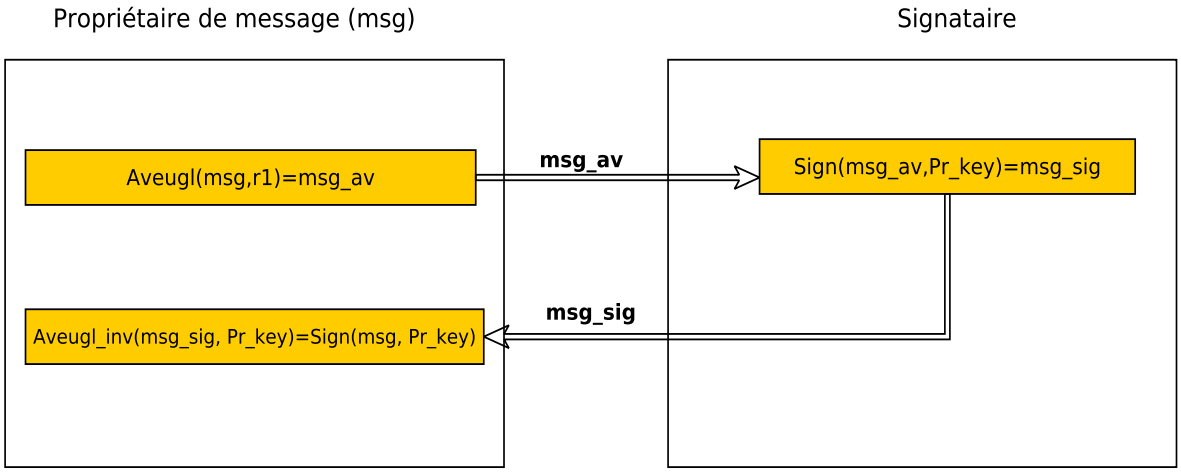


FIGURE 2.5 – Principe de la Signature Aveugle.

2. le signataire utilise sa clé privée Pr_{key} pour signer le message aveuglé msg_{av}, le résultat est le message signé msgs : $sign(msg_{av}, Pr_{key}) = msg_{sig}$.
3. le propriétaire de message utilise la fonction inverse de la fonction d'aveuglement sur le message msg_{sig} pour obtenir une signature du message d'origine msg : $aveugle^{-1}(msg_{sig}, b) = sign(msg, Pr_{key})$.

Notons que ce type de signature est utilisable dans plusieurs applications comme le vote électronique et le transfert anonyme de monnaie électronique.

2.7.2.2 Signature de Groupe

Concept introduit par Chaum et Van Heyst en 1991 [Chaum91]. Cette technique est utilisée généralement pour vérifier l'appartenance d'un individu à un groupe (qui peut avoir un accès à des ressources) sans révéler son identité. Le principe est semblable à celui de la signature numérique classique [Diffie76], du fait qu'il existe une clé privée pour la signature et une clé publique pour la vérification. La différence réside dans le fait que chaque individu « i » appartenant à un groupe « G » possède une clé privée de signature « SKg_i », mais, une seule clé publique de vérification « KV_g » existe. Cette clé permet de vérifier la validité de la signature et par conséquent la vérification de l'appartenance au groupe « G ». Un protocole de signature de groupe passe par les étapes suivantes :

1. l'enregistrement : Pour appartenir à un groupe « G », un individu « i » doit s'enregistrer à ce groupe pour avoir sa clé privée de signature « SKg_i ».
2. la signature : pour signer un message « msg », un individu « i » utilise sa clé privée « SKg_i » le résultat est un message signé msg_{sig} ($sign(msg, SKg_i) = msg_{sig}$).

3. la vérification : pour vérifier la validité de la signature, un vérificateur utilise la clé publique de vérification « VKg » du groupe « G » et le message signé msg_{sig} ($verif(msg_{sig}, VKg)$).

2.7.2.3 Preuve de connaissance à divulgation nulle

La première idée de ce concept est introduite par Goldwass Micah et Rackoff [Golawasser85] puis développé par d'autres chercheurs [Feige88, Goldreich94, Goldreich96]. Cette technique permet à une partie (le prouveur) de prouver à une autre partie (le vérificateur) qu'une déclaration donnée (une connaissance) est vraie, et cela sans transmettre aucune information en dehors de la véracité de cette déclaration. En général, prouver une connaissance nécessite que le prouveur connaît certaines informations secrètes. Par définition, un protocole de preuve de connaissance à divulgation nulle implique que le vérificateur ne sera pas en mesure de prouver la connaissance à son tour à quelqu'un d'autre, puisque le vérificateur ne possède pas l'information secrète. Une explication intuitive de ce principe est introduite dans l'article intitulé « How to explain zero-knowledge protocols to your children » [Quisquater89]. Les auteurs dans cet article utilisent une variante de caverne d'Ali Baba (voir Figure 2.6), où une personne P1 (le prouveur, en violet dans la figure), essaie de prouver à une personne P2 (le vérificateur) qu'elle connaît le mot secret pour ouvrir la porte magique de la caverne.

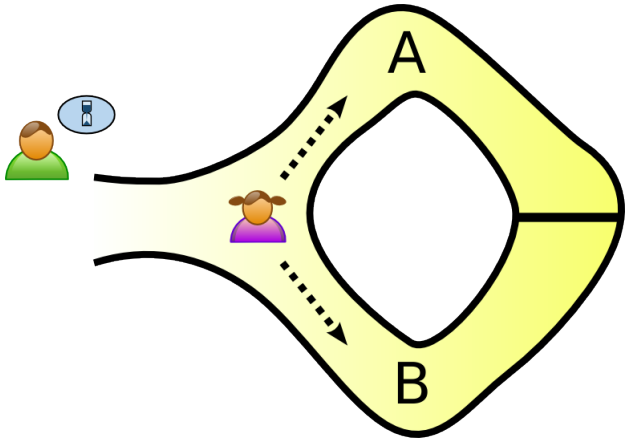


FIGURE 2.6 – Protocole de preuve de connaissance à divulgation nulle- exemple avec la caverne d'Ali Baba [Dake05].

Le protocole de preuve dans cet exemple se déroule comme suit :

1. la personne P2 (le vérificateur) attend à l'extérieur de la caverne que la personne P1(le prouveur) entre.
2. P1 prend le chemin A ou B, sans que P2 le sache.
3. le vérificateur entre dans la caverne et crie un nom de chemin (soit A soit B) que le prouveur doit utiliser pour revenir.

En recommençant plusieurs fois ce protocole, le vérificateur peut collecter suffisamment d'informations pour être quasiment sûr que le prouveur possède ou non le mot secret.

2.7.3 Réseaux de communication anonyme

Ce type de technologies, traite les différentes techniques permettant de communiquer de manière anonyme dans un réseau IP. L'anonymat doit être implémenté pour assuré la protection de l'identité de l'envoyeur et/ou du receveur du message. En plus de l'anonymat, un tel système doit assurer aussi la non-chaînabilité et la non-observabilité (voir section 2.4). Des solutions issues de la sécurité informatique comme les VPNs et les Proxys peuvent être utilisées dans ce type de système. Le problème avec ces solutions c'est qu'elles n'assurent pas toutes les principes de la vie privée (eg. non-chaînabilité et la non-observabilité). Des solutions spécifiques, qui assurent une meilleur protection de la vie privée ont été proposés, parmi lesquelles, nous citons : Mixnets [Chaum81], Tor [Dingledine04] et Crowds [Reiter98].

2.7.3.1 Mixnets

Mix-nets (Mix networks), une notion proposée initialement par Chaum[Chaum81]. Un réseau Mix-net est constitué d'un ou plusieurs Mix-serveur. Chaque mix-server reçoit un ensemble de messages chiffrés à l'entrée et effectue une permutation (mixage) suivie d'une transformation cryptographique à l'aide d'un algorithme de ré-encryptage et/ou de décryptage [Costa17]. Avant qu'un noeud envoie un message sur le réseau Mix-net, il doit choisir l'ensemble des Mix-serveurs à travers lesquelles son message va être redirigé. Il encrypte ensuite le message avec chacune des clés publiques des Mix-serveurs choisis et le redirige vers le premier Mix-serveur. En recevant le message, chaque Mix-serveur le décrypte avec sa clé privée. Ensuite, il fait une permutation aléatoire (le mixage) pour réordonner l'ordre de chiffrement et enfin, il transmet le message au Mix-serveur suivant selon l'ordre de la permutation. La figure 2.7 illustre ce principe de fonctionnement. Ainsi, un réseau Mix-net assure l'anonymat en utilisant les techniques de cryptage, et aussi la non-chaînabilité en cachant le lien existant entre les messages entrants et sortants avec les permutations.

2.7.3.2 Tor

Tor (The Onion Router) [Dingledine04], initialement développé avec la marine américaine au milieu des années 90, Une fois que les militaires passent à des sys-

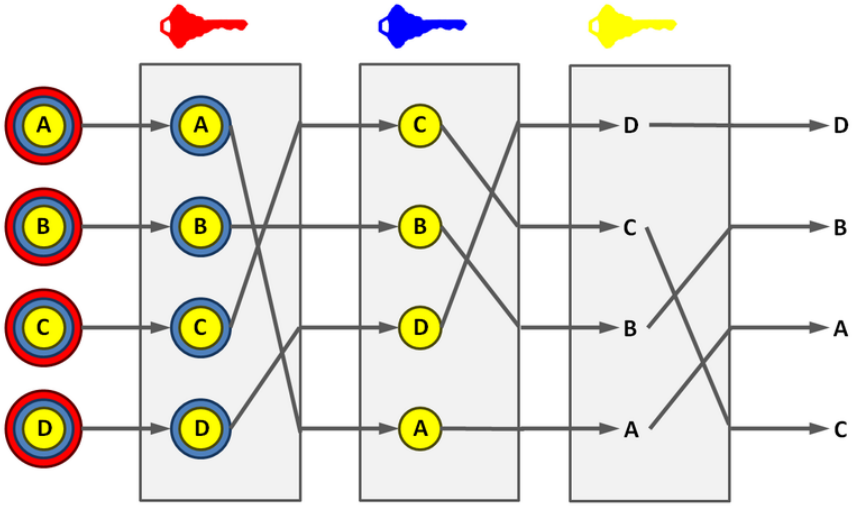


FIGURE 2.7 – Exemple de fonctionnement d'un Mix-net à base de décryptage [Primepq08].

tèmes VPN internes, TOR a été publié en tant que logiciel libre. Actuellement, il est géré par le projet Tor[8], une organisation à but non lucratif, qui défend la confidentialité et la sécurité sur Internet. Actuellement, c'est le réseau de communication anonyme le plus utilisé dans le monde. Le réseau Tor est constitué d'un groupe de serveurs (routeurs onion) gérés par des bénévoles. Son principe de fonctionnement est très semblable à celui des Mix-nets, à part l'étape de mixage qui n'existe pas, d'ailleurs, c'est l'une des raison qui rend Tor plus efficace et plus rapide qu'un réseau Mix-net [Ritter13]. La sécurité dans Tor est basée principalement sur le choix des routes difficiles à prédire par un adversaire. Cependant, et contrairement à Mix-nets, Tor est incapable de résister contre une attaque d'un adversaire capable de voir le chemin entier des serveurs Onion.

2.7.3.3 Crowds

Crowds est un protocole de communication anonyme proposé par Reiter et Rubin en 1998 [Reiter98]. Son objectif est de permettre aux utilisateurs de communiquer avec des serveurs web sans divulguer leurs identités. L'idée derrière Crowds est de se mélanger dans la foule (the crowd en anglais), en d'autres termes, obscurcir les actions d'un individu dans celles d'un groupe, en transmettant aléatoirement les requêtes des membres entre eux avant de les envoyer à leur destination finale. Au niveau architecture, Crowds est un réseau peer-to-peer avec une politique de sélection de chemin différente du Mix-nets et Tor. Le principe de fonctionnement se repose sur l'organisation des utilisateurs dans des groupes appelés Crowds. Chaque utilisateur d'un groupe exécute un processus qui joue le rôle d'un proxy,

8. https://www.torproject.org

appelé Jondo . Quand un Jondo reçoit une requête envoyée par un autre Jondo dans Crowds, il transmet aléatoirement cette requête soit à un autre Jondo avec une probabilité $P_f > 0.5$, soit directement à sa destination (le serveur Web) avec une probabilité $1-P_f$. Ainsi, les serveurs Web, les autres membres du Crowd ainsi qu'un observateur tiers ne peuvent pas déterminer avec précision l'origine de la requête. Chaque Jondo qui transmet un paquet à un autre Jondo enregistre l'identité du Jondo prédécesseur. De cette façon, un tunnel virtuel est construit, qui sera utilisé pour la communication entre l'expéditeur et le récepteur. Comme chaque Jondo ne stocke que des informations sur son prédécesseur, il est impossible pour un Jondo intermédiaire de connaître le chemin entier ni l'identité de l'expéditeur.

2.7.4 Protection à base de langages de spécification des exigences de vie privée

Plusieurs approches basent leur protection de la vie privée sur la mise en correspondance entre les exigences de la vie privée, exprimées par les utilisateurs et les politiques de vie privée spécifiées par les fournisseurs de services [Bernsmed12, Cranor02b, Clement08, Abderahim12, Levy05]. La figure 2.8 illustre le principe de fonctionnement de ce type d'approches. En général, un utilisateur

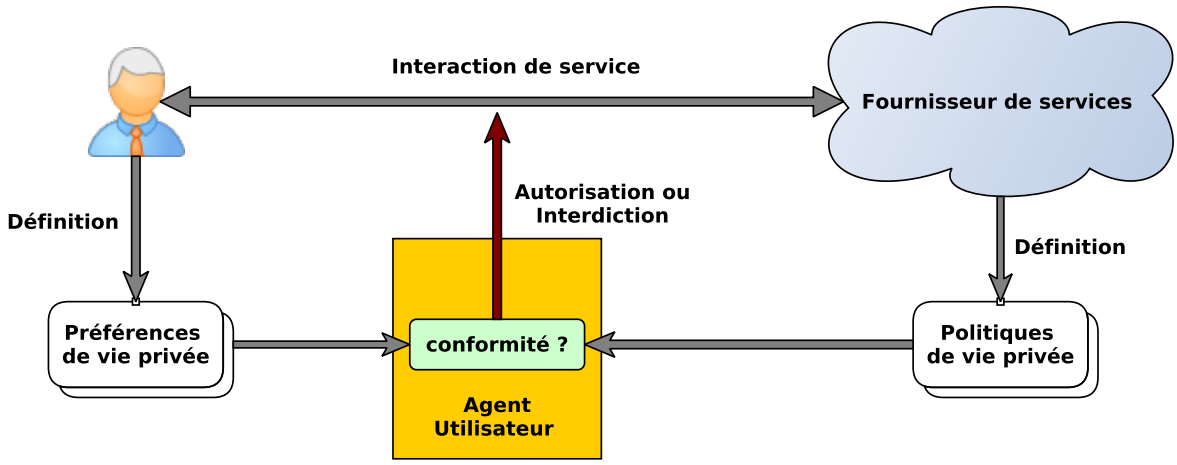


FIGURE 2.8 – Modèle de base pour la mise en correspondance automatique des préférences et la politique de vie privée d'un utilisateur final et d'un fournisseur de services.

utilise un agent de vie privée, son rôle est de vérifier la conformité entre les exigences de vie privée définies par l'utilisateur, et la politique de vie privée définie par le fournisseur de service. Dans le cas positif (une conformité), l'agent utilisateur autorise l'utilisation de service, sinon il y aura une interdiction. Dans ce cadre plusieurs langages de spécification de préférences et de politiques de vie privée. Cette section présente quelques standards, à savoir : la Platform for Privacy Preferences

(P3P) [Cranor02a], A P3P Preference Exchange Language (APPEL) [Langheinrich02], eXtensible Access Control Language (XACML) [Standard05] et l'Enterprise Privacy Authorization Language (EPAL) [Ashley03].

2.7.4.1 Platform for Privacy Preferences (P3P)

La Platform for Privacy Preferences (P3P) [Cranor02a], est un standard développé par le consortium World Wide Web (W3C) [9]. Cette plateforme définit un langage qui permet aux sites Web d'exprimer leurs pratiques en matière d'utilisation des informations recueillies auprès des utilisateurs (politique de vie privée). P3P utilise un format standard codé en XML qui peut être récupéré automatiquement et interprété facilement par les agents utilisateurs. Les agents utilisateurs P3P permettront aux utilisateurs d'être informés des pratiques du site (en format lisible par la machine et par l'homme) et d'automatiser la prise de décision en fonction de ces pratiques. Ainsi, les utilisateurs n'ont pas besoin de lire les politiques de vie privée à chaque site qu'ils visitent. Une politique P3P est une collection de vocabulaire et d'éléments de données qui décrivent les pratiques d'utilisation de ces données. La spécification P3P inclut également un protocole pour la demande et la transmission des politiques P3P. Parmi les critiques majeurs de P3P c'est la complexité du langage de spécification. Cette complexité a donné la possibilité à une politique P3P d'être interprétée et présentée différemment par les différents agents utilisateurs ce qui a limité son adoption par les sites Web.

2.7.4.2 A P3P Preference Exchange Language (APPEL)

Le langage APPEL [Langheinrich02], est un langage développé par le consortium W3C conjointement avec P3P. Ce langage permet à un utilisateur de spécifier ces préférences en termes de vie privée. Ainsi, un Agent P3P sera capable de détecter automatiquement si une politique P3P est conforme aux préférences APPEL spécifiées par l'utilisateur. Une spécification APPEL est constituée d'un ensemble ordonné de règles de préférences de vie privée, appelé « Ruleset ». Une règle APPEL contient un ensemble d'expressions et un comportement. Les expressions définissent un modèle a comparé avec une politique P3P. Si la comparaison est mène à une conformité, la règle est alors déclenchée. Le déclenchement de la règle signifie l'exécution de l'action définie dans la partie comportement de la règle. Du fait que APPEL est étroitement lié à P3P, ils ont subi les même critiques.

9. https://www.w3.org/P3P/

2.7.4.3 eXtensible Access Control Language (XACML)

Le langage XACML [Standard05] défini par OASIS [10] (Advancing Open Standards for the Information Society), et est devenu un standard en 2003. XACML est conçu pour exprimer à la fois les politiques de sécurité et de la vie privée, dans un format lisible par les machines. Plus précisément, XACML permet de faire la spécification des politiques de contrôle d'accès, en définissant des règles de contrôle d'accès structurées en politiques et ensembles de politiques. Ces règles permettent de répondre aux requêtes qui demandent d'effectuer des opérations sur des ressources. La réponse peut être soit positive (permit) soit négative (deny). Chaque règle dans une politique doit définir : Le sujet concerné (personne ou application),les ressources à accéder, les actions demandées, les conditions à satisfaire et la réponse d'une règle dans le cas où les conditions imposées sont satisfaites[Dari Bekara12]. Puisque XACML n'est pas développé spécialement pour la protection de la vie privée, il gère seulement quelques aspects de la vie privée (eg. Le but de collecte des données), de ce fait, il est considéré comme complémentaire de P3P [Anderson04].

2.7.4.4 Enterprise Privacy Authorization Language (EPAL)

EPAL [Ashley03] langage développé par IBM, sa spécification a été soumis au World Wide Web Consortium (W3C) en 2003 pour être considéré pour recommandation. Comme P3P, EPAL est un langage de spécification de politique de vie privée basé sur XML, mais conçu spécialement pour que les entreprises et les organisations spécifient leurs politiques internes de vie privée. Il est aussi un langage d'interopérabilité pour l'échange de politiques de vie privée dans un format structuré entre applications ou entreprises. Les politiques EPAL peuvent être utilisées en interne, mais aussi entre une organisation et ses partenaires commerciaux pour assurer le respect des politiques de vie privée sous-jacentes de chacun [Stufflebeam04]. Le langage EPAL définit un ensemble d'éléments sous forme de listes de hiérarchies, ces éléménts sont : les catégories de données, les catégories d'utilisateurs, d'objectifs de collecte de données, ainsi que des ensembles d'actions, d'obligations et de conditions (de vie privée). Ces éléments sont ensuite utilisés pour formuler des règles d'autorisation de confidentialité qui autorisent ou refusent des actions sur des catégories de données par catégories d'utilisateurs à certaines fins sous certaines conditions tout en imposant certaines obligations [Powers03]. Parmi les critiques majeurs de ce langage, c'est que l'application efficace et correcte des politiques spécifiées dans la couche de stockage de données n'a pas été abordée. Les stratégies spécifiées au niveau EPAL doivent être appliquées au moment de

10. https://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf

l'accès aux données. Dans la plupart des cas, ces données sont stockées dans des bases de données et sont fréquemment consultées. Ainsi, si chaque accès aux données devait reposer sur une évaluation de politique externe, la performance serait inacceptable .

2.8 Conclusion

Dans ce chapitre nous avons présenté une vue générale sur la vie privée sur internet. En particulier, nous avons introduit les principales définitions relatives à la vie privée, un tour d'horizon sur la législation qui cadre et régularise ce domaine, ainsi que les principales menaces et attaques sur la vie privée des utilisateurs, et enfin quelques exemples de technologies et approches de lute contre ces attaques. Il est bien de signaler qu'aucune solution de protection n'est parfaite. À titre d'exemple, les systèmes de gestion d'indenté réduisent la dissémination des informations personnelles et évitent la gestion d'un grand nombre de comptes. Ces même systèmes sont considérés par les Hackers comme un « Big Phish ». Une défaillance dans un tel système peut entraîner de graves conséquences sur les victimes, du fait qu'il gère l'accès à plusieurs systèmes et applications. Les solutions à base des techniques cryptographiques, comme les systèmes d'accréditations anonymes et les réseaux de communication anonyme, sont efficaces en termes de protection d'un côté, mais coûteuses en consommation de ressources et en temps d'exécution d'un autre côté, ce qui pose des problèmes de scalabilité et de temps de réponse. Les approches de protection à base de langages de spécification des exigences de vie privée facilitent pleinement la spécification et la mise en correspondance des préférences et les politiques de vie privée. Malheureusement, les langages de spécification dont ces systèmes se basent, sont souvent complexe, ce qui introduit des ambiguïtés d'interprétation des spécifications faites avec ces langages. Cela a beaucoup limité leur adoption par les sites Web et les fournisseurs de services. Enfin, nous notons que le domaine de la vie privée est un domaine très vaste et très complexe, et loin d'être englober dans sa totalité dans le cadre d'une seul thèse.

"You can have security and not have privacy, but you cannot have privacy without security."

–Tim Mather, Security Expert, Veracode, Inc.

3

Une approche Anti-phishing à base de liste blanche personalisée

▷ *Ce chapitre présente une approche de détection automatique des pages de phishing. L'approche combine deux méthodes : une liste blanche personnalisée et une méthode à base de Machine Learning. La liste blanche est utilisée comme un filtre pour bloquer les pages imitant les pages légitimes usuelles d'un utilisateur. Les pages de phishing qui ne sont pas bloquées au niveau de la liste blanche sont traitées par un classifieur SVM dont nous avons proposé quelques attributs de classification. Les tests effectués sur le système proposé ont montré une amélioration par rapport à d'autres solutions.*

◁

Plan du chapitre

3.1 Introduction

L'hameçonnage (Le phishing) est considéré comme l'une des d'attaques les plus répandus, entraînant des pertes financières importantes pour les entreprises et les utilisateurs. Malgré les efforts pour luter contre ce type d'attaques, les attaques de phishing ne cessent d'augmenter, ce qui influence négativement le fonctionnement des services bancaires, financiers et commerciales sur Internet.

Parmi les solutions anti-phishing on trouve les solutions à base « balack-list/whitelist ». Une liste noire (blacklist)[moz18, Google12, phi18] contient une liste d'URL des sites connus comme phishing. Une page dont l'url existe dans cette liste est bloquée directement. Malgré leur haute précision les blacklists souffrent du problème dit « *zero-day attack* », une fenêtre de vulnérabilité existe chez les utilisateurs avant qu'un site est reconnu comme un site de phishing. Une liste blanche évite ce problème du fait qu'elle contient la liste des URL légitimes, une page dont l'Url n'existe pas dans la liste blanche est qualifiée comme une page suspecte. Le grand inconvénient de cette solution est qu'elle doit contenir tous les sites légitimes, ce qui est impossible et par conséquent un grand nombre de faux positifs. Plusieurs travaux [Cao08, Reddy11, Wang08] ont proposé des améliorations sur les listes blanches, ils ont proposé des listes blanches personnalisées qui contiennent seulement les URL des sites utilisés par un utilisateur particulier, ce qui évite la gestion et la mise à jour de grandes quantités de données. Malgré cela, une liste blanche dans ces travaux repose toujours sur la forte condition : si un Url n'existe pas dans la liste il est suspect, d'où un nombre considérable de faux positifs.

Dans ce chapitre nous présentons une approche à base de liste blanche personnalisée pour la détection automatique des sites de phishing. Contrairement aux travaux précédents notre approche utilise une liste blanche en combinaison avec un classifieur SVM (Support Vector Machines). Les sites de phishing qui ne sont pas bloqués au niveau de notre liste blanche sont traités par un classifieur SVM dont nous avons proposé et amélioré quelques attributs de classification. Les tests effectués sur le système proposé ont montré une amélioration considérable par rapport à d'autres solutions.

Le reste du chapitre est organisé comme suit : la section 3.2 est consacrée à un état de l'art des solutions anti-phishing. La section 3.3 introduit les détails de l'approche proposée. Les résultats des tests seront discutées dans la section 3.4. Enfin, nous concluons et présentons quelques perspectives dans la section 3.5.

3.2 Le phishing : Etat de l'art

3.2.1 Définitions

Selon Le APWG (Anti Phishing Working Group) [Group17] le phishing (hameçonnage en français) est l'acte criminel utilisant à la fois l'ingénierie sociale [soc18] et autres techniques de duperie pour voler des données d'identification personnelles et financières des utilisateurs, telles que noms d'utilisateurs, mots de passe, numéros de sécurité sociale, numéro de cartes de crédits, etc. En général, les phishers (terme désignant l'acteur de l'acte de phishing) dupant les internautes par le biais d'un courrier électronique semblant provenir d'une entreprise de confiance, typiquement une banque ou un site de commerce.

3.2.2 Types d'attaques

On distingue deux principales classes d'attaques de type phishing [Emigh07] : *le phishing à base de malwares (Malware-based phishing)* et *le phishing à base de tromperies (deceptive phishing)* :

- **Le phishing à base de malwares :** Ce type d'attaques se base sur un logiciel malveillant qui exploite les failles de sécurité et s'installe sur la machine de l'utilisateur. Ensuite, ce logiciel malveillant capture des informations confidentielles et les envoie au phisher.
- **Le phishing à base de tromperies :** Dans le phishing à base de tromperie un pirate envoie des e-mails trompeurs semblant provenir d'une institution digne de confiance. Le phisher invite l'utilisateur à cliquer sur un lien qui le redirige vers un site frauduleux pour l'inviter à révéler des informations privées, ces informations sont exploitées par le phisher pour des buts non légitimes. Le phisher dans ce type d'attaques utilise plusieurs techniques, pour tromper l'utilisateur. A. Bergholz & al [Bergholz08] les ont classées en plusieurs types dont on cite :
  - *l'ingénierie sociale* : qui inclut toutes les méthodes et les scénarios inventés par les phishers pour créer un contexte convainquant.
  - *l'imitation* : qui consiste à forger des sites et des liens qui ressemblent au maximum aux sites cibles au phishing.
  - *l'email spoofing* : qui permet à un phisher de falsifier les adresses sources d'un e-mail.
  - *la dissimulation des URL (URL hiding)* : qui permet au phisher de cacher les faux URL vers lesquels les utilisateurs seront redirigés.

O. Salem & al [Salem10] ont présenté d'autres types d'attaques dont nous citons :

— **Pop-up attaque (in-session attack) :** cette technique consiste à lancer un pop-up malveillant devant des sites légitimes en demandant aux utilisateurs de se connecter [ins08], une victime se connecte en croyant que l'origine de ce pop-up est le site légitime adjacent. Un autre type d'attaques qui ressemble pleinement à cette attaque est nommé « Tabnabbing attack » [Aza11], au lieu d'utiliser un pop-up, un site malveillant change complètement son apparence après un certain temps en utilisant un code JavaScript, la nouvelle apparence simule une page de login d'un site couramment utilisé (ex : un Web mail, un réseau social, etc.). En navigant sur d'autre sites un utilisateur peut se tromper et se loge dans le site malveillant.

— **Le Voice phishing :** dans ce type d'attaque la victime reçoit un e-mail de phishing classique, mais au lieu de rediriger la victime à un site web, l'e-mail demande à la victime de fournir des informations personnelles par téléphone.

3.2.3 Les solutions anti-phishing

Plusieurs solutions anti phishing ont été proposées. Ces solutions sont classées selon l'approche utilisée pour lutter contre le phishing. Cette section présente les approches les plus pertinentes, avec quelques exemples de travaux relatifs à chaque approche.

3.2.4 Les solutions à base « Blacklist/whitelist »

Une liste noire (Black list) contient les adresses URL des sites de phishing connus. La solution blacklist est généralement déployée en tant que barre d'outils ou extension de navigateurs Web. Comme exemple d'outils implémentant ce type de solution on trouve : Mozilla Firefox [moz18], google safe browsing [Google12] et phishtank [phi18].

Une liste blanche (whitelist) contient les adresses URL des sites légitimes connus . Plusieurs travaux ont essayé d'améliorer les solutions à base de listes blanches. Les auteurs dans [Reddy11] ont proposé une liste blanche individuelle, cette liste contient les pages de login des connexions usuelles pour un utilisateur, modélisées sous forme d'un ensemble de paramètres. A chaque fois qu'un utilisateur fait une tentative de connexion, les paramètres tirés de cette page serons comparés avec celles de la liste blanche. L'utilisateur est averti si aucune similarité n'est présente. Pour la construction de la liste blanche, les auteurs ont utilisé un classifieur Bayesien pour identifier les pages de connexion légitimes. Dans le même contexte les auteurs dans [Cao08] ont proposé une solution à base de liste blanche, la liste contient les URL de toutes les banques de l'Inde. Une distance est calculée entre L'URL du site visité est les URLs de la liste blanche à l'aide de

l'algorithme *Levenshtein* [Levenshtein65]. Si la distance est inférieure à un seuil (distance minimale), une comparaison à base d'adresse IP est effectuée, si les IP correspondent alors la page est jugée légitime. Dans [Wang08], la liste blanche est conçue pour être remplie par l'utilisateur. La liste est utilisée seulement dans le cas ou l'utilisateur essaie de se connecter sur une page en envoyant des informations sensibles.

3.2.5 Les solutions à base de classification

Ce type de solutions se base en général sur les techniques de classification d'intelligence artificielle. La différence entre les travaux réside dans les attributs utilisés comme entrée des classifieurs. Nous classifions ces solutions selon le sujet de la classification :classification à base d'e-mails, d'URL et à base de contenu des pages.

3.2.5.1 Classification à base d'e-mail

Cette solution classe les e-mails d'une manière similaire aux filtres anti-spam. Les e-mails dans ce cas sont classifiés comme légitimes ou phishing e-mail.Cette solution peut être implémentée au niveau des serveurs mails ou bien au niveau client. [Khonji11] a fait une étude sur les attributs les plus utilisés dans ce type de classification.

PILFER [Fette07], est un exemple d'implémentation d'une telle approche. PILFER utilise un algorithme à bas de forêts aléatoires (Random Forests), et permet de détecter 96% des phishing e-mails, avec seulement 0,1% de faux positifs. Les résultats de cette approche sont améliorés par [Salem10], grâce à l'introduction de nouveaux attributs de classification.

O. Salem & al [Salem10] proposent un système intelligent à base de règles floues. Le système classe les e-mails en trois catégories : les emails sûrs (safe e-mails), les e-mails partiellement sûrs et les phishing e-mails. Dans le cas d'un e-mail partiellement sûrs l'utilisateur est averti. Le système proposé a pu certifier 95% des e-mails légitimes comme sûrs et 5% comme partiellement sûrs.

Dans [Abu-Nimeh07] S. Abu-Nimeh & al, présentent une étude comparative entre plusieurs techniques de classification à base de Machine Learning. Les méthodes comparées sont : la régression logistique (Logistic Regression LR), les arbres de régression et de classification (Classification and Regression Trees : CART), les arbres Bayésiens additifs de régression (Bayesian Additive Regression Trees : BART), Support Vector Machines (SVM), Random Forests (RF), et les réseaux de neurones. Les auteurs ont utilisé 43 attributs pour le modèle et une base de 2889

e-mails (1171 phishing emails et 1718 emails légitimes). A part un petit avantage pour la méthode Random Forests (RF), Les résultats obtenues n'ont pas montré une grande différence entres les diverses méthodes utilisées.

3.2.5.2 Classification à base de contenu

Ce type de solution classe les pages (légitime ou phishing) en évaluant leur contenu. L'exemple le plus cité dans la littérature pour cette approche est CANTINA [Zhang07] : cette solution détecte si un site est un site de phishing à travers l'extraction d'une signature à partir du contenu du site, cette signature est composé de cinq mots obtenus en appliquant l'algorithme TF-IDF (Term Frequency-Inverse Document Frequency). La signature est utilisée comme mot clé dans une requête sur un moteur de recherche. Si ce site fait partie des premiers résultats alors le site est défini comme légitime, sinon, c'est un site de phishing. Cette méthode a pu détecter 90% des sites de phishing, avec 1% de faux positifs. Dans [He11] les auteurs ont utilisé 12 attributs comme paramètre d'entrée d'un classifieur SVM. L'approche a pu détecter 97% de vrais positifs avec 4% de faux positifs.

3.2.5.3 Classification à base d'URL

Ce type de solution est semblable aux précédentes, la seule différence est que les paramètres de classification sont tirés des URL seuls. S.Garera & al [Garera07], ont étudié les caractéristiques des URL des sites de phishing. À partir de cette étude ils ont pu retirer un ensemble d'attributs pour alimenter un classifieur de type logistic regression (LR). Ils ont utilisé pour l'apprentissage et le test une base de 2508 URL dont 1245 phishing URL et 1263 URL légitimes. Cette méthode a pu détecter 95% des URL de phishing avec 1,2% de faux positifs. Un travail très similaire au précédent est ce lui de J.Ma & al. [Ma09], les auteurs ont utilisé trois méthodes de classification : *Naive Bayes*, *Support Vector Machine (SVM)* et *Logistic Regression (LR)*. L'apprentissage et le teste sont fait sur quatre bases différentes. Le meilleur score obtenu dans ce travail est : 0.9% comme taux d'erreur et 0.8% de faux positifs.

3.3 Une approche à base de liste blanche personnalisée

Le plus grand inconvénient d'une liste blanche c'est qu'elle ne peut pas contenir tous les sites légitimes, une telle solution risque de produire un nombre important des faux positifs (un site légitime identifé comme un site de phishing). Une liste blanche personnelle évite ce problème du fait qu'elle contient uniquement les sites utilisés par un utilisateur particulier. Dans notre approche, nous proposons une

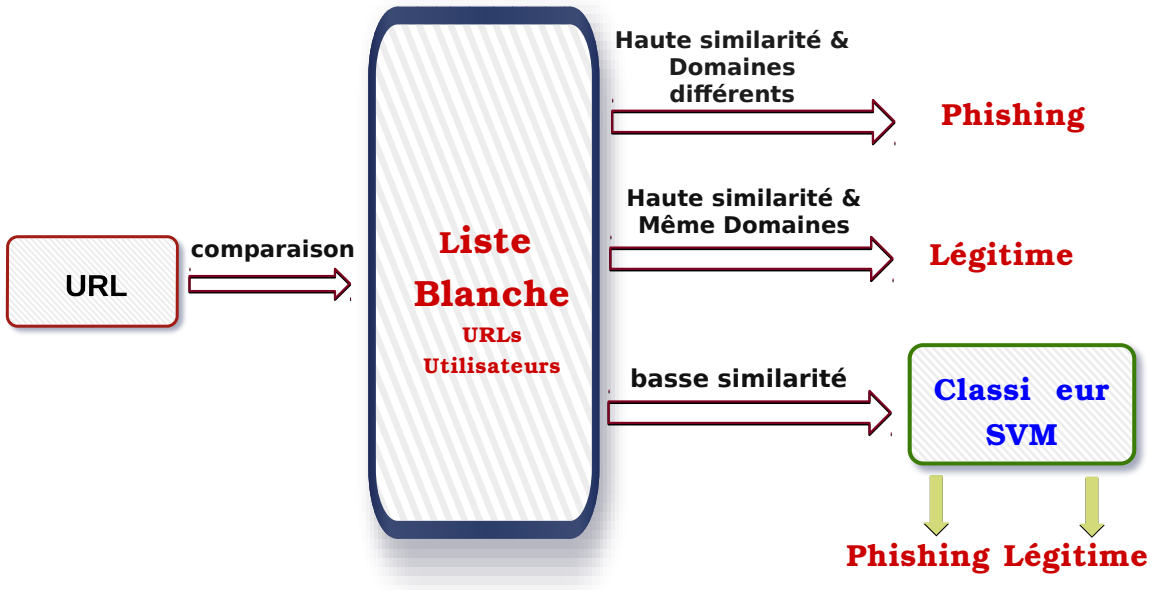


FIGURE 3.1 – L'approche proposée

solution qui combine entre une liste blanche personnalisée et un classifieur SVM. La figure 3.1 montre les deux composants principaux de la solution proposée. Cette solution est censée être implantée comme extension d'un navigateur Web.

Lorsqu'un utilisateur essaie d'accéder à une page, un calcul de similarité est fait entre cette page et les pages de la liste blanche. Selon le degré de similarité trois cas peuvent se produire :

- *Cas 1* : s'il y a une haute similarité *(sim > seuil)* entre la page visitée et l'une des pages de la liste blanche, avec des URLs qui portent des noms de domaine différents, alors la page est considérée comme une page de phishing.
- *Cas 2* : s'il y a une haute similarité *(sim > seuil)* entre la page visitée et l'une des pages de la liste blanche, avec des URLs qui portent le même nom de domaine, alors la page est considérée comme une page légitime.
- *Cas 3* : s'il y a une similarité basse *(sim < seuil)* alors la page est traitée par le classifieur SVM qui décide si c'est une page légitime ou non.

La suite de cette section décrit la structure de la liste blanche ainsi que les attributs utilisés dans le classifieur SVM.

3.3.1 Structure de la liste blanche

La liste blanche dans l'approche proposée est sous forme d'un fichier XML (voir Figure 3.2), qui contient les URL des pages de connexion (login pages) des sites utilisés par l'utilisateur, ainsi qu'un ensemble de mots clés. Les mots clés sont issus à partir de l'arborescence DOM (Document Object Model) des pages, plus précisément :

— Le contenu de la balise "title", ex : `<title> text </title>`.
— Le contenu de la balise "meta keywords", ex : `<meta name ="keywords" content ="text"/>`.
— Le contenu de la balise "meta description", ex : `<meta name = "description" content ="text"/>`.

```
<WhiteList>

<Website>
<url> http://www.facebook.com/</url>
<KeyWords> facebook facebook helps
connect share people life...
</KeyWords>
</Website >
...
<Website >
<url> https://www.paypal.com/</url>
<KeyWords> paypal send money payments
credit credit e-mail ... </KeyWords>
</Website >

<WhiteList>
```

FIGURE 3.2 – La structure de la liste blanche.

Ces trois balises fournissent une description précise du contenu de la page Web. Le texte obtenu à partir de chaque balise passe par une phase d'élimination des mots vides et les caractères de ponctuation. Les mots restants sont concaténés et stockés dans la liste blanche avec l'URL de la page. La figure3.2 donne un exemple du contenu de la liste blanche.

Les mots clés sont utilisés pour le calcul de similarité entre une page visitée et les pages de la liste blanche. Nous avons utilisé un modèle *"sac de mots"* [Salton79] pour la construction des vecteurs de fréquences pour les mots clés de chaque page, et une mesure à base de *"cosinus"* [Singhal01] pour le calcul de similarité. La similarité entre une page visitée "P_v" et une page de la liste blanche "P_l" est calculée comme suit :

$$Cosdis(P_v, P_l) = \frac{\sum_{(t \in Pv) \wedge (t \in Pl)} f_{Pv,t} \cdot f_{Pl,t}}{\sqrt{(\sum_{t \in Pv} f^2_{Pv,t}) . (\sum_{t \in Pl} f^2_{Pl,t})}} \quad (3.1)$$

Où, $f_{x,t}$ est la fréquence du terme "t" dans l'ensemble "x". Deux pages sont similaires si leur score de similarité est proche de 1.

La liste blanche contient en premier lieu les pages de login des 10 sites les plus

attaqués par le phishing [ope10]. Au fur et à mesure s'ajoutent les pages légitimes les plus utilisées par un utilisateur particulier. Cela, évite toute intervention des utilisateurs (qui est souvent une source d'erreurs), et donne une grande simplicité de mise en ouvre et facilite l'installation et l'utilisation.

3.3.2 Les attributs de la classification.

Si la page Web visitée n'a aucune similarité avec les pages de la liste blanche, un vecteur d'attributs est construit pour décrire cette page. ce vecteur est utilisé par la suite par un classifieur SVM pour décider si la page est légitime ou non. Dans notre approche, nous utilisons huit attributs pour représenter une page $P =< A1, A2, A3, A4, A5, A6, A7, A8 >$. Certains attributs sont construits en fonction de l'URL de la page Web et d'autres en analysant son contenu. La section suivante décrit chaque un de ces attributs.

1. **Attribut 1 (A1) : URL avec adresse IP**

   Pour des raisons de minimisation des coûts, de nombreuses pages d'hameçonnage utilisent une adresse IP plutôt qu'un nom de domaine, contrairement à un site légitime auquel on accède le plus souvent par un nom d'hôte. Un URL qui ne contient pas de nom de domaine et qui demande des informations sensibles aux utilisateurs est probablement un site d'hameçonnage. En se basant sur cette remarque, l'attribut $A1$ est définit comme un attribut binaire qui prend la valeur 1 si l'Url est une adresse IP et 0 dans le cas contraire.

2. **Attribut 2 (A2) :l'existence des caractères spéciaux dans l'URL**

   Pour cet attribut nous avons testé l'existence du caractère « @ » dans l'URL. Ce caractère permet de spécifier dans l'URL les paramètres d'accès à un site (ex : nom d'utilisateur et mot de passe). Les phishers utilisent souvent ce caractère pour essayer de tromper les utilisateurs en forgeant des URLs qui ressemblent à des URLs légitimes . Par exemple si un utilisateur rencontre l'URL suivant : http ://paypal.com@www.phish-paypal.com, il peut croire qu'il est sur une page légitime de Paypal. Cet attribut est aussi binaire, il prend la valeur 1 si le caractère « @ » existe dans l'URL et 0 si non.

3. **Attribut 3 (A3) : l'existence d'un certificat SSL (Secure Sockets Layer)**

   La majorité des institutions financières et commerciales possèdent un certificat SSL pour leurs sites Web, ce qui n'est généralement pas le cas pour les sites d'hameçonnage. L'attribut A3 prend alors la valeur 1 en cas d'existence d'un certificat SSL et 0 si non.

4. **Attribut 4 (A4) : l'identité de la page Web est conforme à son URL**

L'identité d'une page signifie le nom de domaine le plus fréquent dans liens existants dans une page. Par exemple, l'identité de la page "*www.facebook.com*" est « *facebook.com* », du fait que dans l'ensemble des liens de la page c'est le domaine qui a la plus haute fréquence. En général, les pages légitimes ont une identité qui correspond à leur URL, et cela malgré l'existence des liens vers d'autres domaines. Le cas des pages de phishing est différent, une page qui imite une autre page légitime garde en général les mêmes liens que cette dernière, ce qui produit une identité différente du domaine de l'URL de la page. Notre attribut A4 prend alors la valeur 1 si l'identité de la page correspond au domaine de son URL, et 0 si non.

5. **Attribut 5 (A5) : moteur de recherche**

   Une variante de cet attribut est utilisé par [Zhang07] et [He11]. Le principe est d'utiliser la technique TF-IDF (term frequency-inverse document frequency) sur une page Web pour extraire une signature de cette page (les mots les plus pertinents). La signature est utilisée comme mots clés dans un moteur de recherche. Si l'URL de la page est parmi les N premiers résultats de la recherche alors la page est jugée légitime, si non c'est une page contrefaite. Nous avons utilisé le même principe, mais au lieu d'appliquer la technique TF-IDF, nous avons utilisé la même méthode d'extraction des mots clés utilisée dans la génération de la liste blanche (section 2.3.1). Les mots clés de la requête utilisée sur le moteur de recherche se composent des quatre mots les plus fréquents parmi les mots résultants de la phase d'extraction, plus l'identité de la page. Pour la recherche, nous avons utilisé le moteur "metacrawler [1]", un méta moteur qui renvoie les résultats pertinents de trois moteurs de recherches (google, yahoo et bing). L'attribut A5 prend alors la valeur 1 si l'URL de la page est parmi les 20 premiers résultats du moteur de recherche, et 0 si non.

6. **Attribut 6(A6) : pourcentage des liens vides**

   Un lien vide est un lien qui ne pointe vers aucune page. On général, un lien vide est spécifié par les balises suivantes : `<a href="#">`, `<a href="javascript::void(0)">`. Certains pages de phishing imitent une page légitime en remplaçant les liens pointant vers des pages externes par des liens vides. Un grand pourcentage des liens vides rend une page suspecte. L'attribut non binaire A6 est calculé comme suit :

$$F6 = \frac{L_N}{L_T} \text{ If } L_T > 0; \quad F6 = 0 \text{ If } L_T = 0. \tag{3.2}$$

   Où : L_N est le nombre des liens vides, et L_T est le nombre total des liens dans une page.

1. http ://www.metacrawler.com

7. **Attribut 7 (A7) : la fréquence des liens**

   Certaines pages d'hameçonnage utilisent des images au lieu de code html pour imiter l'apparence d'un site Web légitime, réduisant ainsi le nombre de liens pointant vers d'autres pages. Cet attribut permet de calculer la fréquence des liens pointant vers d'autres pages par rapport aux liens qui pointent vers des images ou des scripts, et est calculée comme suit :

$$F7 = \frac{L_P}{L_T} \text{ If } L_T > 0; \;\; F7 = 0 \text{ If } L_T = 0. \tag{3.3}$$

   Où : L_P est le nombre des liens pointant vers des pages, et L_T est le nombre total des liens.

8. **Attribut 8 (A8) : Action conforme à l'identité de la page**

   Une page de connexion demande des informations d'accès aux utilisateurs à l'aide d'un formulaire contenant des champs de saisie (balise "input"), comme le montre l'exemple suivant :

```
<form method="post" action="action.php">
<input name="login" id="username" />
<input name="passwd' id="passwd" />
<button type="submit" > Connexion </button>
</form>
```

   Les informations entrées dans les zones de saisie (champ Input) sont traitées par une fonction dont l'URL est spécifiée dans la zone action de la balise "form" . Habituellement, les pages Web d'hameçonnage revendiquent une identité de page légitime, mais le champ d'action contient une URL différente par rapport à cette identité. L'attribut trinaire A8 modélise ce comportement comme l'algorithme 1 le présente :

3.4 Évaluation

Pour évaluer notre approche, nous allons tester la validité de la liste blanche, puis les performances du classifieur SVM.

3.4.1 La liste blanche

Pour évaluer la performance de la liste blanche, nous avons utilisé 400 pages, dont 200 légitimes et 200 pages phishing. Les pages légitimes sont issues de :

Algorithme 1 : Le calcul de l'attribut A8

```
1  if toutes les actions de la page sont conformes avec son identité then
2  |  A8 = 1;
3  else
4  |  foreach forme avec le champ action qui ne correspond pas à l'identité de la
   |  page do
5  |  |  if il existe un Input de type password dans cette forme then
6  |  |  |  if il existe un lien réciproque entre l'URL de l'action et l'identité de la
   |  |  |  page then
7  |  |  |  |  A8 = 1;
8  |  |  |  else
9  |  |  |  |  A8 = 0;
10 |  |  |  |  return ;
11 |  |  |  end
12 |  |  else
13 |  |  |  A8 = −1;
14 |  |  end
15 |  end
16 end
```

— Les pages de connexion (login) des 10 sites Web les plus visés par le phishing [ope10].
— Les pages de connexion de 50 sites parmi les sites les plus visités selon Alexa. [2].
— 140 pages à partir de Yahoo Random [3].

Toutes les 200 pages de phishing sont collectées à partir de PhishTan [4].

Comme nous l'avons déjà mentionné, la liste blanche contient dans un premier temps que les informations relatives aux pages de connexion des 10 sites les plus visés par le phishing [ope10].

Pour choisir un seuil adéquat, nous avons testé notre liste blanche avec trois valeurs : 0.7, 0.8 et 0.9. Le tableau 3.1 résume les résultats du test.

Les résultats montrent que plus le seuil de similarité est élevé, plus les mauvaises décisions de la liste blanche sont basses. Avec un seuil de 0.7, la liste blanche détecte plus de pages de phishing (36 pages contre 34 et 30 pour les seuils 0.8 et 0.9 respectivement). Par contre, il y a un plus grand nombre de décisions incorrectes (2 pages légitimes ont été classées comme pages de phishing, contre 1 et 0 pour les seuils 0.8 et 0.9 respectivement). Comme nous cherchons à éviter toute classification incorrecte au niveau de la liste blanche, nous avons adopté une valeur de 0.9 pour notre seuil de similarité. Avec cette valeur, la liste blanche a détecté environ 5% de toutes les pages légitimes et 15% des pages d'hameçonnage.

2. http ://www.alexa.com/topsites
3. http :// random.yahoo.com/bin/ryl
4. http ://www.phishtank.com/phish_archive.php

Seuil	PagesWeb(PW)	Phish détecté	Legitim détecté
≥ 0.7	Legitim PW (200)	02	10
	Phishing PW (200)	36	0
≥ 0.8	Legitim PW (200)	01	10
	Phishing PW (200)	34	0
≥ 0.9	Legitim PW (200)	00	10
	Phishing PW (200)	30	0

TABLE 3.1 – Résultats d'évaluation de la liste blanche.

Malgré ce faible pourcentage, nous notons l'absence de classifications incorrectes (c'est-à-dire que le taux de faux positifs est de 0%).Nous rappelons que les pages à faible similarité ne sont pas traitées au niveau de la liste blanche, ce qui contribue fortement à ce faible taux de filtrage.

Enfin, nous notons que l'efficacité de la liste blanche s'accroîtra davantage au fur et à mesure de l'utilisation. Les pages les plus utilisées par un utilisateur vont s'ajouter automatiquement à la liste blanche, cela diminue pleinement le risque qu'un utilisateur soit victime d'une page de phishing qui imite une de ces pages usuelles.

3.4.2 Le classifieur SVM

Si une page a une faible similarité avec les pages de la liste blanche, cette page est transformée en un vecteur d'attributs. Ce dernier sera utilisé par un classifieur pour décider si la page est légitime ou non. Dans notre nous avons utilisé un classifieur SVM [Chang11], un classifieur binaire très adapté à notre cas puisque nous disposons que de deux classes (phish ou légitime).

Nous avons utilisé une base de 850 pages. 400 pages sont celles utilisées pour le test de la liste blanche, 200 sont des pages légitimes de Yahoo Random, et 250 sont des pages de phising collectées auprès de PhishTank.

400 pages sont utilisées pour l'apprentissage (200 légitimes et 200 pages de phishing), les 450 pages restantes sont consacrées aux tests (200 pages légitimes et 250 pages de phishing).

Avant de transformer la base de test en vecteurs d'attributs, nous avons appliqué la liste blanche comme filtre. La liste blanche a pu détecter 41 pages de phishing et 0 pages légitimes. Les 409 pages restantes sont transformées en vec-

teurs d'attributs et transmises au classifieur SVM. Notons que toutes les pages de phishing ont été transformées en vecteurs d'attributs au moment ou elles sont encore onlines.

Nous avons évalué notre modèle de classification en fonction de :

— Taux de vrais positifs (TP aussi appelé rappel) : pourcentage des pages d'hameçonnage classées correctement.

— Taux de faux positifs (FP) : pourcentage des pages légitimes classés à tort comme pages d'hameçonnage.

— La précision (P) : la mesure dans laquelle les pages identifiées comme pages d'hameçonnage sont effectivement malveillantes.

— Le F-measure (FM) : la moyenne harmonique entre la précision et le rappel.

Ces différentes mesures sont calculées comme suit :

$$TP(R) = \frac{P_P}{P_P + P_L} \tag{3.4}$$

$$FP = \frac{L_P}{L_P + L_L} \tag{3.5}$$

$$P = \frac{P_P}{P_P + L_P} \tag{3.6}$$

$$FM = 2 \times \frac{P \times R}{P + R} \tag{3.7}$$

Où :

P_P est le nombre de pages Web d'hameçonnage correctement classées ; P_L est le nombre de pages d'hameçonnage classées incorrectement comme légitimes ; L_P est le nombre de pages légitimes classées à tort comme pages d'hameçonnage ; L_L est le nombre de pages légitimes classés correctement.

Le tableau 3.2 résume nos résultats d'évaluation sur l'ensemble des données de test susmentionnés.

	TP	FP	P	FM
Values	98%	3.5%	96.6%	97.3%

TABLE 3.2 – Résultats d'évaluation sur la base de test (performances du classifieur SVM)

Nous avons comparé notre modèle de classification à celui de CANTINA [Zhang07] and [He11](l'approche de M. He et al.).

La figure 3 montre une légère amélioration de notre modèle en termes de taux de vrais positifs : notre modèle a pu détecter 98% des pages d'hameçonnage contre 89% pour [Zhang07] et 97% pour [He11]. Si on compte les 41 pages détectées au niveau de la liste blanche (puisque les pages ont été correctement classées), les

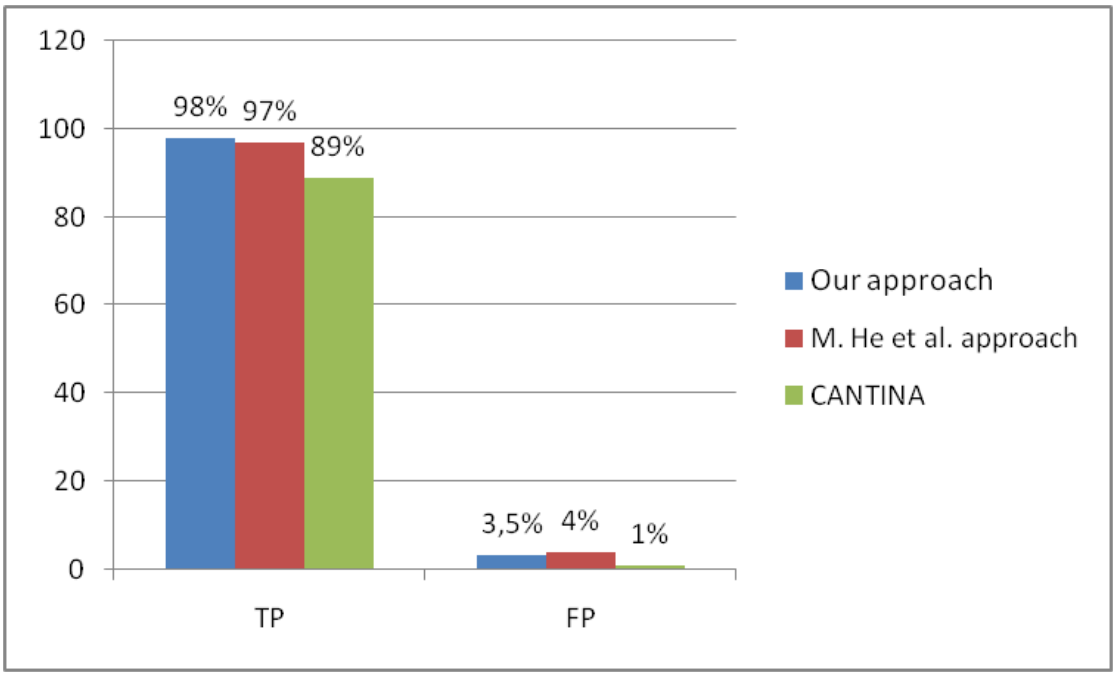


FIGURE 3.3 – Comparaison avec des travaux antérieurs.

résultats s'améliorent encore, comme le tableau 3.3 montre.

	TP	FP	P	FM
Values	98.4%	3.5%	97.2%	97.7%

TABLE 3.3 – Résultats d'évaluation sur la base de test(SVM+WHITLIST).

Notre approche souffre d'un taux élevé de faux-positifs (> 3%) ; cette valeur élevée est principalement associée au classifieur SVM. Comme mentionné précédemment, l'application du classifieur SVM diminuera après la stabilisation de la liste blanche, du fait que la liste blanche sera incrémentée progressivement par l'historique de navigation de l'utilisateur ; à ce stade, la plupart des décisions de classification serons prises au niveau de la liste blanche et le risque d'une attaque de phishing réussie contre l'utilisateur diminue considérablement puisque les pages d'hameçonnage qui tentent d'imiter une page fréquentée par l'utilisateur seront détectées par la liste blanche.

Si une page d'hameçonnage imite une page non incluse dans la liste blanche, il est peu probable qu'un utilisateur fournisse à ce site des informations sensibles, et même dans le cas où l'utilisateur fournit de telles informations ; la page sera détectée comme une tentative d'hameçonnage par notre classifieur SVM avec une forte probabilité.

3.5 Conclusion

Dans ce chapitre, nous avons décrit une solution anti-phishing qui combine une liste blanche personnalisée et un moteur de classification automatisé. Notre approche combinée bénéficie des avantages des deux techniques sans souffrir des inconvénients de chaque méthode.

Nous avons maintenu l'exactitude des solutions à base de listes blanches en éliminant la difficulté de gérer et mettre à jour de grandes quantités de données en utilisant une liste blanche personnalisée. Les faux positifs traditionnellement présents dans ces solutions sont aussi éliminés, du fait que si une page ne fait pas partie de la liste blanche, elle n'est pas classée comme une page de phishing mais traitée par notre classifieur SVM. De plus, la liste blanche proposée est conçue pour être automatiquement mise à jour sans intervention de l'utilisateur, ce qui réduit considérablement les erreurs de configuration de l'outil et facilite pleinement son utilisation.

Malgré tous ses avantages, l'approche proposée souffre de certaines inconvénients. Par exemple, notre approche est incapable de détecter si des sites Web légitimes sont attaqués par une usurpation DNS (DNS spoofing). Nous pouvons remédier à ce problème en ajoutant les adresses IP de chaque page à la liste blanche, du fait que les adresses IP de la majorité des sites visés par le phishing sont souvent stables [Cao08]. Un autre inconvénient c'est la dépendance d'un des attributs du modèle de classification à un moteur de recherche. Cela peut affecter la réactivité de l'outil en cas de dysfonctionnement du moteur de recherche.

Enfin, les performances de notre approche peuvent être améliorées, les attributs de classification ont besoin d'être optimisés pour un fonctionnement meilleur, tandis que de nouvelles attributs pertinents peuvent être découverts à l'avenir pour différencier davantage les pages légitimes et d'hameçonnage.

"When it comes to privacy and accountability, people always demand the former for themselves and the latter for everyone else."

–David Brin

4

Une approche de préservation de la vie privée pour la sélection de services Web composites

▷ *Les technologies à base des services Web sont considérées parmi les technologies les plus prometteuses pour l'intégration des sources de données hétérogènes, et la réalisation des opérations complexes. La question de la protection de la vie privée des utilisateurs dans les services Web demeure l'une des préoccupations majeures liées à ce domaine. Dans ce chapitre nous présentons une approche de sélection qui préserve les exigences de vie privée des services Web, tout en minimisant le risque induit par l'utilisation des données privées par ces services.* ◁

Plan du chapitre

4.1 Introduction

De nos jours, les services Web jouent un rôle central dans le fonctionnement d'Internet. Grâce à leur flexibilité et à leur modularité, ils sont utilisés comme la technologie principale de communication et d'échange de données dans des modèles logiciels récents comme le cloud computing. L'un des avantages les plus importants de l'utilisation d'une telle technologie réside dans sa capacité à composer des services web existants. Cela permet de créer des nouvelles fonctionnalités avec moins d'efforts et de ressources.

En raison du nombre croissant de services sur Internet, la sélection de services Web devient une tâche cruciale. En bref , la tâche de sélection consiste à choisir parmi un ensemble de services candidats, ceux qui répondent le mieux aux besoins des utilisateurs. Généralement, la sélection des services Web repose sur des critères non fonctionnels couvrant diverses catégories telles que la qualité de service (QoS), la sécurité ou la confidentialité. Trouver un service ou une composition de services qui répondent le mieux aux exigences non fonctionnelles est connu pour être un problème NP-difficile [Alrifai12]. Pour cette raison, la conception de systèmes de séléction efficaces reste un problème de recherche très important.

Bien que la majorité des approches de sélection de services Web sont basées sur la qualité de service (QoS) comme principal critère de sélection [Alrifai12, Wu12, Huang09, Yu07, Liu13, Halfaoui15]„ les préoccupations croissantes à l'égard de la protection de la confidentialité des données ont donné lieu à de nombreuses études traitant la problématique de protection de la vie privée dans les services Web [Ke13, Ke15, Tbahriti14, Costante13, Guermouche07, Xu06, Kwon11, Rezgui02, Squicciarini13, Carminati15].
Malgré ces efforts, le concept de vie privée demeure un problème difficile dans ce domaine. En effet, il implique de nombreuses autres questions telles que : i) Comment représenter les politiques de confidentialité afin de permettre aux utilisateurs et aux fournisseurs de services de spécifier leurs besoins en matière de protection de la vie privée, ii) Comment gérer efficacement les politiques spécifiées pour trouver une composition qui satisfait toutes les contraintes de confidentialité , iii) Si une telle composition n'existe pas, comment assurer un niveau acceptable de protection de la vie privée.

Notre travail aborde le problème de protection de la vie privée dans le contexte de la sélection de services Web. Notre objectif est de protéger à la fois les utilisateurs et les fournisseurs de services contre la violation de la vie privée. Cette dernière, peut être causée par une mauvaise utilisation ou une divulgation non autorisée des données échangées. Pour assurer cet objectif, nous proposons un Framework de sélection de la vie privée. Ce Framework vise à trouver un service Web composite

qui préserve les exigences de confidentialité des utilisateurs et des fournisseurs de services. Le Framework prend en entrée un plan d'échange de données de composition, un ensemble d'exigences de confidentialité des utilisateurs ainsi que les politiques de confidentialité de chaque service candidat. La sortie est une composition concrète qui répond le mieux à toutes les contraintes de confidentialité en entrée.

La contribution de ce travail est comme suit :

— Premièrement, nous proposons deux modèles de protection de vie privée : un *modèle de composition privée* et un *modèle de politique de vie privée*. Le modèle de composition est utilisé pour représenter le plan d'échange de données entre les services d'une composition. Dans ce modèle, seules les données qualifiées comme privées sont prises en compte. Le modèle de politique de vie privée est utilisé pour exprimer les préférences et les exigences en termes de vie privée des utilisateurs et les services. Dans ce modèle, la définition de la vie privée est fondée sur quatre dimensions : l'objectif, la visibilité, la granularité et le temps de rétention. Ces quatre dimensions sont considérées comme les prédicats les plus complets pour définir les besoins en matière de protection de vie privée. De plus, l'utilisation de ces dimensions rend notre modèle compatible avec la norme P3P du framework de confidentialité w3c[Cranor02a, Ghazinour11].

— La deuxième contribution consiste à définir une fonction de divulgation de données à base d'intégrale floue [Grabisch96]. Cette fonction mesure le risque de menace à la vie privée, causée par les fournisseurs de services lors de l'utilisation des données privées. Dans notre modèle de confidentialité, cette fonction est utilisée pour classer les compositions qui satisfont les exigences de confidentialité, ce qui nous permet de sélectionner la composition qui porte le risque de menace minimal.

— Basée sur les modèles de confidentialité proposés, notre troisième contribution propose trois algorithmes de sélection. L'objectif principal est de démontrer la faisabilité et la compatibilité de nos modèles avec les différents types d'algorithmes. Le premier algorithme est une adaptation d'un algorithme de type meilleur d'abord que nous appelons CBFS (Constrained Best First Search). L'adaptation est faite pour gérer les contraintes de vie privée. Les autres algorithmes sont basés sur deux des modèles déclaratifs les plus populaires : la Satisfaisabilité booléenne (SAT)[Biere09] et le Answer Set Programming (ASP)[Janhunen16]. Dans ces derniers modèles, la spécification du problème est codée comme une formule propositionnelle ou un programme ASP. Le code est utilisé par la suite par un solveur pour rechercher les solutions. L'efficacité de ces algorithmes est testée et comparée sur différents types de jeux de données.

Le reste de ce chapitre est structuré comme suit : La section 4.2 est consacrée à un état de l'art sur les travaux réalisés dans ce domaine. La section 4.3 fournit une description détaillée du Framework de sélection proposé. Cette description inclut des détails sur le modèle de composition ainsi que sur le modèle de politique de confidentialité. Un formalisme du problème de sélection privée des services Web est présenté dans la section 4.4. La section 4.5 est consacrée aux détails de l'implémentation de la fonction de risque ainsi qu'aux algorithmes proposés. Les expérimentations et les résultats de l'évaluation sont présentés à la section 4.6. Enfin, la section 4.7 conclut et fournit quelques perspectives.

4.2 Travaux connexes

Un bon nombre d'approches ont été consacrées à la protection de la vie privée des services Web. La grande majorité d'entre elles sont basées sur la vérification de la conformité entre les politiques de confidentialité des fournisseurs de services et les exigences de vie privée des utilisateurs. Dans cette section, nous mentionnerons brièvement les approches les plus pertinentes à notre travail. Une analyse plus complète de l'état de l'art est donnée dans [Ke13, Ke15].

Dans [Tbahriti14], les auteurs ont proposé une approche pour améliorer la confidentialité des services Web DaaS (données en tant que service). Dans ce contexte, ils ont proposé un modèle formel permettant aux utilisateurs et aux fournisseurs de services de définir un ensemble de règles de confidentialité pour exprimer leurs politiques et exigences de vie privée. Ils proposent un algorithme appelé PCM (Privacy Compatibility Matching) pour vérifier la compatibilité de la confidentialité entre les politiques et les exigences dans une composition DaaS. Contrairement à notre travail, cette approche décrit un mécanisme de négociation qui établit une réconciliation dynamique entre les services en cas d'incompatibilité. Cependant, les auteurs n'ont utilisé aucune heuristique dans leurs algorithmes, ce qui peut poser des problèmes de passage à l'échelle.
Dans [Costante13], les auteurs ont proposé une approche basée sur les objectifs pour protéger la confidentialité des utilisateurs et les fournisseurs de services dans les compositions de services Web. Le principe repose sur un modèle qui permet aux politiques et aux préférences de confidentialité des utilisateurs et les fournisseurs de services d'être implémenter sur plusieurs dimensions. Contrairement à notre travail, la granularité n'est pas utilisée comme une métrique de confidentialité, les auteurs ont utilisé la sensibilité des attributs de données conjointement avec le but, la visibilité et le temps de rétention. Un algorithme de composition de service est proposé pour vérifier la conformité entre les exigences de confidentialité des utilisateurs et les politiques de confidentialité des fournisseurs de services, puis une

sélection pour choisir la composition qui préserve le mieux la vie privée est faite. La sélection est basée sur une fonction de classement qui agrège les dimensions de confidentialité utilisées. L'algorithme proposé a l'avantage de traiter à la fois les aspects fonctionnels et non fonctionnels (vie privée), tandis que la plupart des approches les traitent en étapes distinctes. Néanmoins, les auteurs de ce travail n'ont pas fourni une réelle implémentation pour prouver leur modèle de vie privée.

Dans [Guermouche07], les auteurs ont proposé un protocole de remplacement pour les services Web en tenant compte la protection de la vie privée. Afin de définir ce protocole, les auteurs ont proposé un modèle basé sur des règles qui étendent la norme P3P. Le modèle est utilisé pour exprimer les politiques de confidentialité et les préférences des utilisateurs et les fournisseurs de services. L'intégration de ce modèle dans les protocoles métier a permis une analyse de la remplaçabilité dans les aspects fonctionnels et non-fonctionnels (vie privée). Contrairement à notre travail, le modèle proposé ne prend pas en charge la protection de la vie privée dans le contexte des compositions de services.

Dans [Xu06], les auteurs ont proposé un Framework qui fournit aux consommateurs un service composite respectant la vie privée. Dans ce Framework, le fournisseur de services interagit avec les utilisateurs et fournit les principes (modèle) qui guident l'utilisation des données privées. En cas de conflit entre les préférences du consommateur et le modèle reçu, le consommateur peut assouplir ses politiques de confidentialité afin que le service puisse être utilisé ou plutôt, génère des obligations qui représentent la violation de la vie privée et les transmet au service composite. La principale différence entre cette approche et notre travail est que cette proposition se concentre uniquement sur la protection de la vie privée des utilisateurs et ne traite pas la vie privée des fournisseurs de services. Une autre différence est que le modèle proposé n'utilise que la sensibilité des attributs de données comme prédicat de protection . Cependant, ce travail tient compte du niveau de confiance des fournisseurs de services, ce qui n'est pas le cas de notre travail.

Dans [Kwon11], les auteurs ont présenté une approche de négociation pour trouver un compromis entre les préférences des utilisateurs en matière de protection de la vie privée et les exigences des fournisseurs de services. L'approche est basée sur une structure en treillis de Galois. Le treillis est généré à l'aide d'une matrice d'évaluation qui relie les éléments (nom, adresse, e-mail, etc.) et les différents types de sous-services (connexion, inscription, paiement, etc.). Les treillis générés sont ensuite utilisés pour développer un ensemble de règles qui définissent les préférences et les exigences des utilisateurs et des fournisseurs de services. Le processus de négociation se fait par une variation d'un paramètre "θ" sur une plage de 1 à 5 qui a pour effet de modifier le treillis correspondant et donc les règles de confidentialité. L'inconvénient majeur de cette approche réside dans la définition de

la matrice d'évaluation qui reste une tâche difficile pour un simple utilisateur. cette approche ne prend pas en charge la protection de la vie privée des compositions de services.

Dans [Rezgui02], les auteurs traitent la problématique de la vie privée dans les services Web relative aux e-gouvernements. Dans ce cadre, Les auteurs ont proposé trois niveaux de protection de vie privée : la vie privée de l'utilisateur, la vie privée du service et la confidentialité des données. Plus précisément, les entités qui demandent l'accès aux données ou aux opérations d'autres entités doivent fournir des justificatifs d'identité pour exécuter cette fonction. Si l'accès demandé est autorisé, un filtre de données est utilisé pour contrôler l'accès, après quoi, un agent mobile de protection de la vie privée est utilisé pour livrer les données demandées, en s'assurant que le demandeur ne viole pas les exigences de l'entité locale en matière de protection de vie privée. Notons que dans ce travail, les auteurs se sont concentrés uniquement sur la confidentialité des services Web simples et n'ont pas considéré les composition de services.

Dans [Squicciarini13], les auteurs ont proposé un Framework de sélection de services Web dont l'objectif est de protéger les besoins de confidentialité des utilisateurs et les fournisseurs de services. Ce travail répond aux problèmes de confidentialité associés à la divulgation des données personnelles de l'utilisateur lors de la localisation des services Web dans la phase de vérification des règles de approvisionnement des fournisseurs. La confidentialité des règles d'approvisionnement des fournisseurs de services est également couverte. Le principal élément du Framework proposé est le négociateur privé. Ce dernier est basé sur un protocole de correspondance qui permet à deux parties de calculer conjointement l'intersection de leurs entrées, sans divulguer d'informations supplémentaires. Ce travail diffère de notre travail dans le sens que le mécanisme de protection de la vie privée utilisé repose sur des techniques cryptographiques et non sur des règles de confidentialité.

Dans [Carminati15], les auteurs ont proposé une approche de préservation de la vie privée dans les compositions de services Web. Dans cette approche, les auteurs ont proposé un composant appelé ElitePicker application (EPApp). Ce composant implémente un protocole de sécurité basé sur des techniques de chiffrement. Le mécanisme de cryptage dans ce module permet de vérifier de manière privée les besoins des utilisateurs / fournisseurs de services. Contrairement à notre travail qui utilise un modèle d'orchestration , ce travail porte sur le modèle chorégraphie pour la composition de services.

4.3 Un Framework de sélection à base de critères de vie privée

Dans ce chapitre, nous décrivons un Framework de sélection de services qui préserve la vie privée. L'objectif est de trouver une composition de services qui répond aux contraintes de confidentialité des utilisateurs ainsi que les fournisseurs de services.

4.3.1 Architecture du Framework

Le principal composant de ce Framework (Figure 4.1) est le gestionnaire de vie privée. Ce gestionnaire génère un service Web composite (une composition) concret qui répond le mieux aux contraintes de vie privée des utilisateurs et les fournisseurs de services. Pour générer cette composition concrète, le gestionnaire de vie privée utilise un algorithme de sélection, qui prend comme entrée :

— Une composition abstraite qui spécifie le flux de données et le flux de contrôle des services composants (tâches abstraites) ;
— Un ensemble de services concrets qui implémentent les tâches abstraites précédentes, en plus de la spécification de leurs contraintes de confidentialité (stockées dans le registre des services) ;
— Une requête de l'utilisateur qui spécifie ses exigences de confidentialité.

L'algorithme de sélection implémenté dans le gestionnaire de vie privée est basé sur deux modèles : un *modèle de composition* et un *modèle de vie privée*. Le modèle de composition se concentre principalement sur le flux de données qui relie les services d'une composition, tandis que le modèle de vie privée décrit comment définir les règles de confidentialité utilisées pour exprimer les exigences et les contraintes de vie privée. les détails sur ces deux modèles sont donnés dans les deux sections suivantes.

4.3.2 Le Modèle de Composition

Dans notre travail, un service Web composite (composition des services) C est représenté par un couple $C =< D, W >$, où D représente le modèle de flux de données (Data-flow) de la composition et W représente le modèle de flux de traitement (Work-flow). Dans un contexte de protection de la vie privée, nous nous intéressons plus au modèle de flux de données, qui représente le plan d'échange des données entre les services élémentaires d'une composition. Le modèle de flux de données est représenté par un graphe orienté valué $D = (S, A)$. Dans ce graphe, l'ensemble des sommets S représente les services élémentaires, alors que l'ensemble des arcs

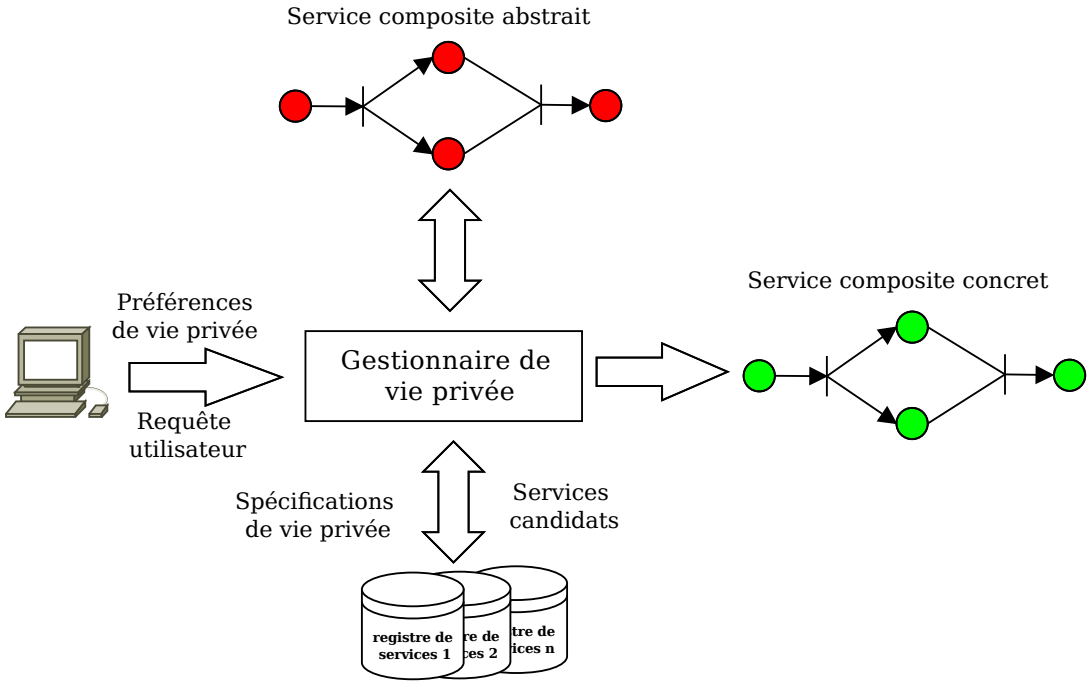


FIGURE 4.1 – Architecture du Framework de sélection privée.

A représente les dépendances de données entre les services. La figure 4.2 représente un simple graphe de flux de données. La figure montre que chaque service de la composition nécessite un ensemble d'attributs (données) pour fournir le service attendu. Par exemple, le service s_3 a besoin des attributs a_1, a_4 et a_5 en entrée pour fournir la fonctionnalité de service. Notons que le service s_1, qui n'a pas d'attributs d'entrée, représente un agent utilisateur. Pour assurer la protection de la confidentialité, chaque service de la composition doit respecter les contraintes de confidentialité requises par les services qui fournissent les attributs nécessaires.

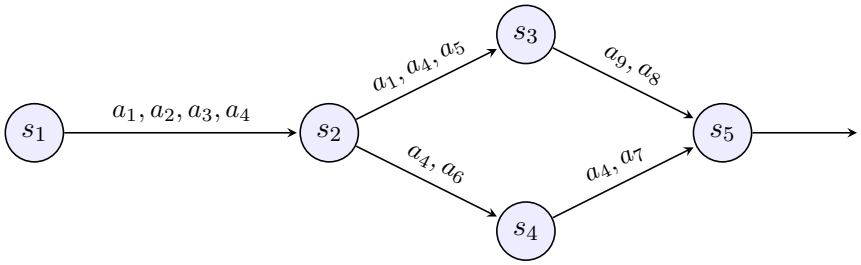


FIGURE 4.2 – Exemple d'un graphe de flux de données.

Nous observons dans cet exemple (voir Figure 4.2) que les attributs a_1 et a_4 (arc (s_2, s_3)) sont fournis par le service s_2 alors qu'ils sont produits par le service s_1. En d'autres termes, le service s_1 est le *propriétaire* des attributs a_1 et a_4. Par consé-

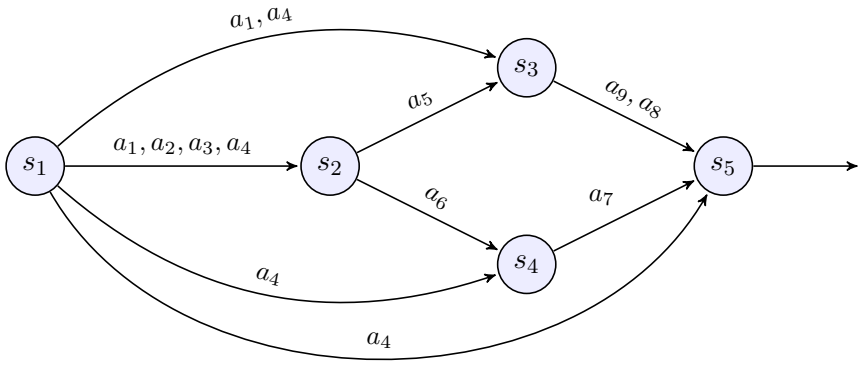


FIGURE 4.3 – Graphe de vie privée correspondant au graphe de flux de données de la figure 4.2

quent, pour ces deux attributs, le service s_3 doit respecter les exigences de confidentialité de s_1 et non celles de s_2.

Selon le graphe de flux de données et le concept de *propriétaire d'attributs*, nous pouvons construire un nouveau graphe appelé *graphe de vie privée*, ce graphe est défini comme suit :

Définition 1 (Le graphe de vie privée). *Un graphe de vie privée d'un graphe de composition $D = (S, A)$ est un graphe orienté valué $P_G = (S, A_p)$, où l'ensemble des arcs A_p représente les contraintes de vie privée liées aux attributs échangés entre services.*

La figure 4.3 représente le graphe de vie privée issu du graphe de composition de la figure 4.2. Par exemple, sur cette figure, l'arc entre s_1 et s_2, signifie que les politiques de vie privée du service s_2 relatives aux attributs a_1, a_2, a_3 et a_4 doivent être conformes aux exigences de vie privée du service s_1.

Un graphe de vie privée est dit **valide** si toutes les contraintes de vie privée entre les noeuds (services) de ce graphe sont satisfaites.

Définition 2 (Ensemble de Précédences de Service.). *Étant donné un graphe de vie privée $P_G = (S, A_p)$, l'ensemble de précédences PRD^{s_x} d'un service $s_x \in S$ est défini comme suit :*

$$PRD^{s_x} = \{s_y \in S \,|\, \exists\, (s_y, s_x) \in A_p\}$$

Exemple : dans le graphe de vie privée de la figure 4.3, l'ensemble de préséances du service s_5 est : $PRD^{s_5} = \{s_1, s_3, s_4\}$.

Définition 3 (Ensemble de dépendances). *Étant donné un graphe de vie privée $P_G = (S, A_p)$, l'ensemble des dépendances DEP^{s_x, s_y} entre deux services $s_x, s_y \in S$ est la valeur de l'arc (s_x, s_y).*

Exemple : dans le graphe de vie privée de la figure 4.3, nous avons : $DEP^{s_1,s_2} = \{a_1,a_2,a_3,a_4\}$, $DEP^{s_2,s_5} = \emptyset$.
Les deux concepts de *l'ensemble de précédences de service* et de *l'ensemble de dépendances* sont utilisés dans la spécification du *modèle de vie privée*, introduite dans la section 4.3.3.

4.3.2.1 Exemple illustratif

La figure 4.4 représente un scénario simplifié dérivé d'un service web d'achat en ligne. Ce service implique trois services élémentaires plus d'un agent utilisateur. Les services élémentaires sont : le service de vente, de paiement et de livraison.

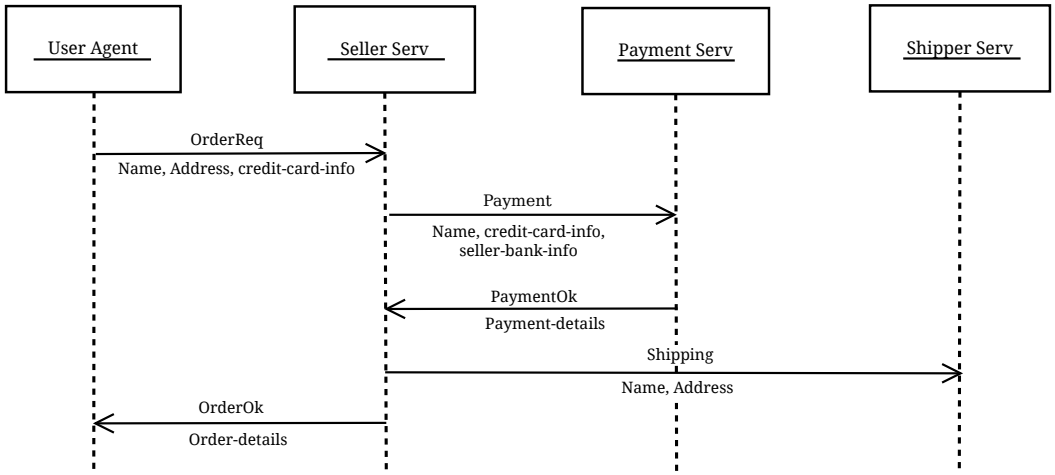
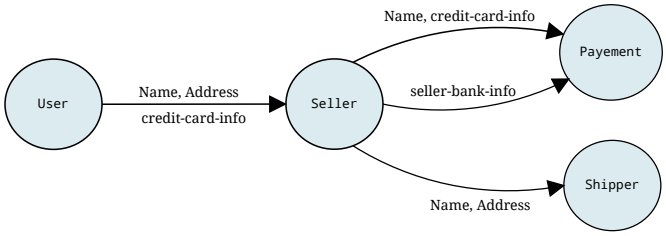


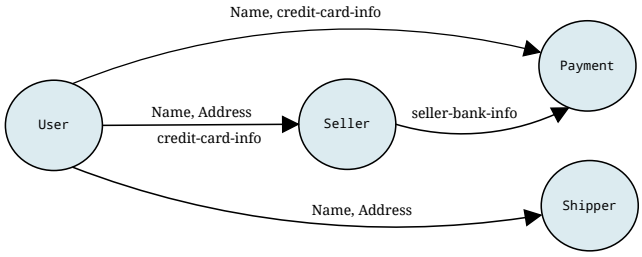
FIGURE 4.4 – Un scénario d'une transaction réussie d'achat en ligne.

Ce scénario décrit une transaction d'achat en ligne réussie. L'agent utilisateur lance la transaction en envoyant au service de vente un message de commande (*OrderReq*) avec les attributs de nécessaires (*Nom, Adresse, credit-card -info*). Ensuite, le service de vente envoie au service de paiement un message de paiement (*Payment*) ainsi que les attributs : *seller-banking-info*, le *Nom* et le *credit-card -info* de l'utilisateur. Dans le cas d'un paiement réussi, le service de paiement informe le service de vente par un message (*PaymentOk*) ainsi que les détails de paiement. Après cela, le service de vente invoque le service de livraison en donnant les détails de cette livraison (Nom et Adresse).

Dans cet exemple, les attributs *nom*, *adresse*, *credit-card-info* (de l'utilisateur) et *vendeur-banque-info* (du vendeur) sont considérés comme privé. La figure 4.5 montre le graphe de flux de données (figure 4.5 (a)) et le graphe de vie privée



(a)



(b)

FIGURE 4.5 – Graphe de flux de données (a) et le graphe de vie privée correspondant (b) du scénario de composition de la figure 4.4

correspondant (figure 4.5 (b)). Notons que dans ces graphes, seuls les attributs jugés comme privés qui sont représentés et toutes les autres données (par exemple *Payment-details*, *Order-details*) sont omises.

Le graphe de vie privé précise les contraintes de vie privée que les trois services doivent vérifier pour réussir une transaction. Par exemple, la politique de vie privée du service de vente doit être conforme aux exigences de vie privée de l'utilisateur, et cela, pour les attributs, *nom*, *adresse* et *credit-card-info*.

À partir de graphe de vie privée, on définit *l'ensemble des précédences*(PRD^s) et *l'ensemble des dépendances* (DEP^{s_1,s_2}) comme suit :

$PRD^{User} = \emptyset$, $PRD^{Seller} = \{User\}$, $PRD^{Payment} = \{User, Seller\}$,

$PRD^{Shipper} = \{User\}$.

$DEP^{User,Seller} = \{name, address,$ **credit-card-inf**$\}$, $DEP^{Seller,Payment} = \{$**seller-bank-inf**$\}$,

$DEP^{User,Payment} = \{address,$ **credit-card-inf**$\}$, $DEP^{User,Shipper} = \{name, address\}$.

4.3.3 Modèle de politique de vie privée

4.3.3.1 Prédicats de vie privée

Pour que les utilisateurs et les fournisseurs de services puissent exprimer leurs politiques et exigences en termes de vie privée correspondant à chaque attribut de données (nom, date de naissance, SSN, etc.), nous avons besoin d'un ensemble de prédicats (ou critères) de confidentialité qui représentent autant que possible toutes les facettes de la vie privée. Dans notre travail, nous avons utilisé le même ensemble de prédicats définis dans [Ghazinour11, Ghazinour14]. Ces prédicats représentent la vie privée comme un point à quatre dimensions. Chaque dimension représente une facette différente de vie privée. Ces prédicats sont : *l'objectif (ou but)*, *la visibilité*, *la granularité* et *le temps de rétention*. Ces prédicats sont définis comme suit :

- **L'objectif (Obj)** :définit comment les données peuvent être utilisées une fois collectées ;
- **La visibilité(V)** : définit qui est autorisé à voir les données fournies ;
- **La granularité (G)** : définit le degré de la précision des données fournies ;
- **Le temps de rétention (T)** : définit la durée de conservation des données par le collecteur de données.

Notons qu'en général, les valeurs que peuvent prendre ces prédicats sont représentés sous forme de treillis ou des structures hiérarchiques[Ghazinour11]. Dans notre travail, nous avons normalisé les prédicats : *visibilité*, *granularité* et *temps de rétention* pour être représentés comme des nombres réels dans l'intervalle $[0..1]$. L'objectif de cette normalisation est de faciliter les comparaisons et le calcul sur ces prédicats.

Pour expliquer le processus de normalisation, nous utilisons le simple exemple de la Figure 4.6. La figure 4.6 (a) représente un exemple d'hiérarchie du prédicat «granularité» corespondant à l'attribut «état civil», tandis que la figure 4.6 (b) représente une simple hiérarchie du domaine du prédicat «visibilité». Comme la figure 4.6 le montre, la normalisation consiste à attribuer une valeur comprise entre 0 et 1 à chaque niveau de la hiérarchie du domaine. Cette affectation est faite de telle sorte que le niveau le plus révélateur de l'information prenne la valeur la plus proche de 1. La valeur normalisée est obtenue en utilisant la formule suivante :

$$N_p = \frac{D_{max} - d_p}{D_{max}} \tag{4.1}$$

Où D_{max} est la profondeur maximale de la hiérarchie de domaine du prédicat, et d_p est le niveau de la valeur correspondant à ce prédicat.

Dans l'exemple de la figure 4.6 (a), si un utilisateur divorcé ne veut pas révéler la valeur exacte de son état civil, mais juste qu'il est marié une fois, la valeur nor-

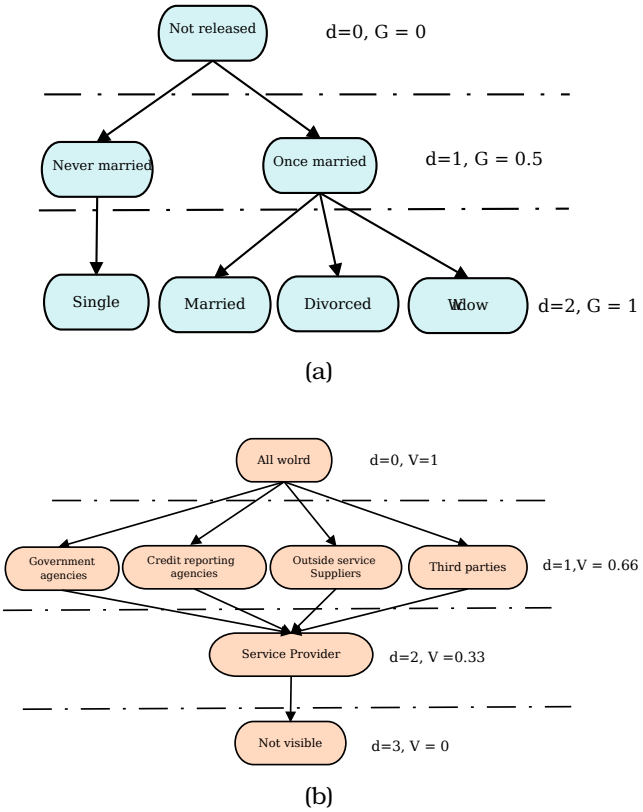


FIGURE 4.6 – Exemple de normalisation des prédicats : Granularité (a) et Visibilité (b).

malisée du prédicat «granularité» de l'état civil sera d 0,5 ($D_{max} = 2$ et $d_p = 1$). De même, et selon la hiérarchie du prédicat «visibilité» de la figure 4.6 (b), si un utilisateur veut rendre un attribut donné visible uniquement au fournisseur de services, la valeur de prédicat «visibilité» liée à cet attribut sera 0.33 ($D_{max} = 3$ et $d_p = 2$).

La normalisation du prédicat «temps de rétention» est plus simple. En fait, il suffit de diviser la valeur du temps de rétention de l'attribut, par la valeur maximale. Notons que pour faire éviter les utilisateurs de gérer des valeurs numériques, on utilise une interface graphique qui leur permet d'exprimer les différentes valeurs de prédicats de vie privée. La correspondance à l?intervalle [0,1] sera effectué automatiquement par le Framework de sélection.

4.3.3.2 Règles de Vie Privée

Une règle de vie privée est une assertion utilisée par les utilisateurs ou les fournisseurs de services pour spécifier leur politique et leurs exigences en termes de vie privée. Une règle de vie privée est définie par les prédicats susmentionnés : Objetif (*Obj*), Visibilité (*V*), Granularité (*G*) et Temps de rétention (*T*). Plus formellement, une règle de vie privée R_i est un quintuplet : $R_i = \langle a_i, Obj_i, V_i, G_i, T_i \rangle$, où :

— $a_i \in Att$, est un attribut de données privées ex : Nom, SSN, âge, adresse, etc.
— Les valeurs : Obj_i, V_i, G_i et T_i représentent les valeurs des dimensions de vie privée : Objectif, Visibilité, Granularité et Temps de rétention.

Une règle de vie privée $R_i = \langle a_i, Obj_i, V_i, G_i, T_i \rangle$, signifie que les attributs privées a_i sont collectés pour l'objectif Obj_i, avec une visibilité V_i, une précision G_i et un temps de rétention T_i. Pour une règle donnée $R_i = \langle a_i, Obj_i, V_i, G_i, T_i \rangle$, nous supposons la syntaxe suivante : $R_i[Att] = a_i, R_i[Obj] = Obj_i, R_i[V] = V_i, R_i[G] = G_i, R_i[T] = T_i$.

Définition 4 (Règles bien définies). *Un ensemble de règles de vie privée P est dit bien défini, si pour toutes les règles de P on ne trouve pas deux règles avec les mêmes valeurs d'attribut, d'objectif et de visibilité, mais des valeurs différentes de granularité et/ou de temps de rétention. Formellement :*

$$\forall R_i, R_j \in P : \begin{cases} \Big((R_i[Att] = R_j[Att]) \wedge \\ (R_i[Obj] = R_j[Obj]) \wedge \\ (R_i[V] = R_j[V]) \Big) \Rightarrow \\ \Big((R_i[G] = R_j[G]) \wedge (R_i[T] = R_j[T]) \Big) \end{cases} \tag{4.2}$$

Intuitivement, une règle de vie privée R_i est caractérisée par un identifiant qui se compose de la structure (a_i, Obj_i, V_i) (ceci est similaire à la clé primaire dans les bases de données relationnelles). Ainsi, toute règle qui contient un triplet (a_i, obj_i, V_i) doit être unique. Sinon, nous obtiendrons un ensemble incohérent de règles. Par exemple, pour le graphe de la Figure 4.5 (b), si un utilisateur définit un ensemble de règles qui contient les deux règles de vie privée suivantes (R_1 et R_2), cet ensemble n'est pas bien défini (incohérent) :
$R_1 = \langle name, shopping\_purpose, service\_provider, all\_released, 60 \rangle$;
$R_2 = \langle name, shopping\_purpose, service\_provider, all\_released, 50 \rangle$.

4.3.3.3 Politiques et Exigences de Vie Privée

Politique de vie Privée (PP) : C'est un ensemble *bien défini de règles* de vie privée définies par chaque fournisseur de services, spécifiant l'ensemble des pratiques de vie privée applicables à toute donnée collectée. Formellement, une politique de vie privée PP^{s_x} d'un service s_x est définie comme suit :

$$PP^{s_x} = \{R_i \in P \mid R_i[Att] \in I^{s_x}\}. \text{ où, } I^{s_x} \subseteq \{\textit{Input de } s_x\} .$$

Par exemple, le service de vente (seller) dans le graphe de vie privée de la figure 4.5 (b), a : $I^{seller} \subseteq \{$*name, address, credit-card-inf*$\}$.
Ainsi, le service de vente doit spécifier un ensemble de règles de vie privée PP^{seller}, qui définit comment il va utiliser les attributs appartenant à I^{seller}. Par exemple,

si le service de vente veut que : i) toute la partie $address$ de l'utilisateur doit être divulguée (G), ii) l'attribut $address$ est collecté pour les objectifs (Obj) : $shopping$ et $statistics$, iii) l'attribut $address$ est partagée avec des tiers, iv), et l'attribut $address$ est conservée pendant 90 jours ; alors, l'ensemble PP^{seller} doit contenir les règles de vie privée suivantes :
$R_1 = \langle address, shopping\_purpose, third\_parties, all\_released, 90\rangle$;
$R_2 = \langle address, statistics\_purpose, third\_parties, all\_released, 90\rangle$.

Exigence de vie privée (PR) : C'est un ensemble *bien défini de règles* de vie privée, spécifiant l'ensemble des conditions de vie privée qu'un fournisseur de service doit respecter lors de l'utilisation des données collectées. Formellement une exigence de vie privée, PR^{s_x}, d'un service s_x est définie comme suit :

$$PR^{s_x} = \{R_i \in P | R_i[Att] \in O^{s_x}\}.\ \text{où},\ O^{s_x} \subseteq \{\textit{Output de } s_x\}\ .$$

Dans le graphe de vie privée de la figure 4.5 (b), le service du vente a :
$O^{seller} \subseteq \{$*seller-bank-info*$\}$. Si le service de vente est prêt à révéler toutes les parties de l'attribut *seller-bank-info*, uniquement au service de paiement et uniquement à des fins de paiement, et exige que cet attribut ne soit pas conservé plus d'un jour, alors, l'ensemble PR^{seller} doit contenir la règle suivante :
$R = \langle$*seller-bank-info*$, payment\_purpose, service\_provider, all\_released, 1\rangle$;

4.3.3.4 Règles comparables

Deux règles de vie privée sont comparables, si elles sont associées au même attribut et si les valeurs de leur prédicat "objectif" appartiennent au même ensemble. Formellement, les deux règles R_i et R_j sont comparables si :

$$\Big(R_i[Att] = R_j[Att]\Big) \wedge \Big((R_i[Obj], R_j[Obj]) \in Gl \times Gl\Big).$$

où, $Gl \subseteq Obj$, représente l'ensemble des objectifs d'une composition donnée.

Par exemple, dans le service d'achat en ligne de la figure 4.4, l'ensemble Gl peut être : $Gl = \{shopping\_purpose, payment\_purpose, shipping\_purpose\}$. Si l'utilisateur définit la règle relative aux exigences de vie privée $R_1 = \langle name, Shopping\_purpose, 0.5, 0.33, 0.2\rangle$, et le service de vente, définit la règle de politique de vie privée $R_2 = \langle name, statistic\_purpose, 0.6, 0.2, 0.2\rangle$. Les deux règles R_1 et R_2 sont incomparables, puisque, $statistic\_purpose \notin Gl$. Par conséquent, nous pouvons affirmer que ces deux règles ne sont pas conformes sans comparer les valeurs des autres prédicats.
Nous définissons la fonction $comp(R_i, R_j)$ qui renvoie 1 si les deux règles (R_i, R_j) sont comparables.

$$comp(R_i, R_j) = \begin{cases} 1 & \text{Si } \Big(R_i[Att] = R_j[Att]\Big) \wedge \Big((R_i[Obj], R_j[Obj]) \in Gl \times Gl\Big) \\ 0 & \text{Sinon} \end{cases} \tag{4.3}$$

4.3.3.5 Règles conformes

Une règle de vie privée R_i est dite conforme avec règle de vie privée R_j ($R_i \sim R_j$) si :

1. Les deux règles sont comparables ;
2. Les valeurs : visibilité, granularité et temps de rétention de R_i sont supérieurs ou égales à celles de R_j .

Formellement :

$$R_i \sim R_j \Leftrightarrow \begin{cases} comp(R_i, R_j) = 1 & \wedge \\ R_i[V] \geqslant R_j[V] & \wedge \\ R_i[G] \geqslant R_j[G] & \wedge \\ R_i[T] \geqslant R_j[T] \end{cases} \tag{4.4}$$

4.3.3.6 Services conformes

Un service s_x qui définit une exigence de vie privée $PR^{s_{xy}}$ est conforme à un service s_y qui définit une politique de vie privée $PP^{s_{xy}}$, si la fonction $Nconf(s_x, s_y)$ renvoie zéro. Cette fonction représente le nombre de règles de vie privée non conformes entre les deux services s_x et s_y. Cette fonction est définie comme suit :

$$Nconf(s_x, s_y) = \begin{cases} 0 & \text{Si } \forall\, R_i \in PR^{s_{xy}}, \exists\, R_j \in PP^{s_{xy}} : R_i \sim R_j \\ k & \text{Sinon , } (k = |NC|) \end{cases} \tag{4.5}$$

où, $NC = \{R_i \in PR^{s_{xy}} | \nexists R_j \in PP^{s_{xy}} : R_i \sim R_j\}$. $PR^{s_{xy}} \subseteq PR^{s_x}$, $PP^{s_{xy}} \subseteq PP^{s_y}$ et définies par : $PR^{s_{xy}} = \{R_i \in PR^{s_x} | R_i[Att] \in DEP^{s_x,s_y}\}$, $PP^{s_{xy}} = \{R_i \in PP^{s_y} | R_i[Att] \in DEP^{s_x,s_y}\}$, où, PR^{s_x} et PP^{s_y} représentent respectivement les ensembles des exigences et des politiques de vie privée définies par les services s_x et s_y, alors que DEP^{s_x,s_y} représente l'ensemble des dépendances entre les services s_x et s_y (voir la définition 3).

4.3.3.7 Service Valide

Un service s_x est dit valide, s'il est conforme à tous les services dont il dépend. Formellement :

$$vld(s_x) = \begin{cases} 1 & \text{Si } \forall s_y \in PRD^{s_x} : Nconf(s_y, s_x) = 0 \\ 0 & \text{Sinon} \end{cases} \tag{4.6}$$

où, PRD^{s_x} Représente l'ensemble de précédences du service s_x (voir la définition 2).

4.3.3.8 Composition Valide

Une composition $C = \{s_{1,}s_2, \cdots, s_n\}$ est dite valide si tous ses services composants sont valides. Nous définissons la fonction $vld(C)$, qui retourne 1 si la composition C est valide :

$$vld(C) = \begin{cases} 1 & \text{Si } \forall s_x \in C : vld(s_x) = 1 \\ 0 & \text{Sinon} \end{cases} \tag{4.7}$$

4.3.3.9 Fonction de Risque relative à la Vie Privée

Puisque l'objectif de notre framework de sélection est de trouver une composition qui préserve toutes les contraintes de la vie privée et minimise le risque lié à une menace relative à cette dernière, nous devons définir une fonction qui mesure la quantité de ce risque. Cette fonction sera utilisée pour classer les compositions qui remplissent les exigences de vie privée, afin de sélectionner la composition avec le risque minimal.

Notons que notre modèle impose de fortes contraintes sur le prédicat « Objectif ». Comme nous avons mentionné précédemment, une règle R_i n'est pas conforme à une règle R_j que si les valeurs du prédicat objectif de ces deux règles appartiennent au même ensemble Gl (voire les sections 4.3.3.4 et 4.3.3.5). Par conséquent, la fonction de risque de vie privée est exprimée uniquement sur les trois prédicats : *visibilité*, *granularité* et *temps de rétention*. Pour définir le risque de vie privée induit par une composition globale, nous devons d'abord spécifier la fonction de risque pour chaque service.

Risque d'un Service Pour définir la fonction de risque de vie privée d'un service s_x, nous devons définir une fonction qui agrège les valeurs des trois prédicats de vie privée V, G et T de chaque règle de l'ensemble PP^{s_x}. Avec l'existence d'une telle fonction, le risque d'une menace à la vie privée induite par un service s_x dans une composition est exprimé par :

$$risk(s_x) = \frac{1}{\sum_{i=1}^{i=|PP|} \lambda_i} \sum_{R_{i=1}}^{R_{i=|PP|}} \lambda_i \times Agr(R_i[V], R_i[G], R_i[T]) \tag{4.8}$$

Où, Agr représente une fonction d'agrégation, et λ_i représente le degré de sensibilité de l'attribut $R_i[Att] = a_i$ de la règle R_i . Notons que le degré de sensibilité est une valeur entre 0 et 1, définie par le propriétaire de l'attribut de données (utilisateur ou fournisseur de services).
La fonction d'agrégation utilisée dans notre travail est définie dans la section 4.5.1.

Risque d'une Composition Étant donné une composition de n services $C = \{s_1, s_2, ..., s_n\}$, le risque global de vie privée de cette composition est obtenu en agrégeant les risques de ses services composants. L'agrégation dépend du Work-flow (le plan d'exécution) de la composition. La table 4.1 présente des exemples d'agrégations en fonction du plan d'exécution.

Plan d'exécution	**Fonction d'agrégation**
Séquentielle	$\sum_{i=1}^{n} risk(s_i)$
Parallèle	$\sum_{i=1}^{n} risk(s_i)$
Boucle	$risk(s_i)$
Conditionnelle	$\sum_{i=1}^{n} p_i \times risk(s_i)$

TABLE 4.1 – Exemples de risque d'une composition en fonction du plan d'exécution.

Comme la table 4.1 mentionne, le plan d'exécution d'une composition (séquentielle, parallèle, en boucle, conditionnelle) n'a pas une grande influence sur le calcul du risque total d'une menace induit par cette composition. Dans la plupart des cas, la fonction d'agrégation est la somme des risques liés à la vie privée introduits par les services composants. L'exception est faite pour la structure conditionnelle, où un seul service s'exécute parmi un ensemble (selon la valeur d'une condition). Dans ce cas, on peut évaluer le risque de vie privée introduit par cette structure comme la somme pondérée des risques de chaque service. Les coefficients de pondération représentent la probabilité de participation de chaque service dans l'exécution. Dans une structure en boucle, l'exécution du même service plus d'une fois n'introduira pas de risque de supplémentaire. Comme nous l'avons déjà mentionné, cette fonction mesure le taux de risque potentiel relatif à la vie privée. Par conséquent, plus la fonction de risque est élevée, plus la protection de la vie privée devient petite.

4.4 Problème de Sélection des services Web avec Préservation de la vie Privée (PSWPP)

4.4.1 Formulation du Problème

Étant donné une composition abstraite $C_A = \{S_1, S_2, \cdots, S_n\}$ représenté sous forme de graphe de vie privée, une liste de services candidats pour chaque classe S_i de C_A et des exigences de vie privée de l'utilisateur U_{req} (un ensemble de règles de vie privée). L'objectif est de trouver une composition concrète $C_c = \{s_1, s_2, \cdots, s_n\}$, où chaque service concret $s_i \in C_c$ est l'implémentation du service abstrait $S_i \in C_A$, tel que :

— La composition $\{U_{req}\} \cup C_c$ est valide.

— La fonction de risque de vie privée est minimisée.

Selon le modèle de vie privée susmentionné, nous devons trouver une composition concrète C_c, avec :

$$C_c = \{s_1, s_2, ..., s_n\} : \begin{cases} vld(\{U_{req}\} \cup C_c) = 1. \\ minimise(risk(C_c)). \end{cases} \tag{4.9}$$

4.4.2 Complexité du Problème de Sélection des services Web avec Préservation de la vie Privée

Le problème général de sélection des services Web avec préservation de la vie privée (PSWPP), avec $n$classes de service (n services abstraits) et k services candidats par classe (k services concrets par classe), peut être formellement représenté comme suit :

$$\begin{cases} \text{minimize } \sum_{i=1}^{n} \sum_{j=1}^{k} s_{ij} r_{ij} \\ \text{subject to } \sum_{i=1}^{n} \sum_{j=1}^{k} s_{ij} v_{ij} \leq V \\ \sum_{i=1}^{k} s_{ij} = 1 \, , \; 1 \leq i \leq n \\ s_{ij} \in \{0,1\} \, , \; 1 \leq i \leq n, \; 1 \leq j \leq k \end{cases} \tag{4.10}$$

où : s_{ij} est le service j de la classe i ; r_{ij} représente le risque de vie privée associé à s_{ij} ; v_{ij} représente le nombre de règles non conformes du service s_{ij}, et V représente le nombre de règles non conformes tolérées par la solution. Notons que le problème formulé dans l'équation 4.9 est un cas particulier de PSWPP (représenté dans l'équation 4.10), où $V = 0$.

Le *PSWPP est un problème PN-hard*. La preuve est faite en réduisant le problème de sac à dos à choix multiple (MCKP :Multiple Choice Knapsack Problem)- connu pour être NP-hard-[Pisinger95] à PSWPP.

Le MCKP est défini comme suit : étant donné N groupes d'items $(G_1, \cdots, G_N)$, chacun d'eux contient k items, c'est-à-dire : $G_j = \{I_{1j}, \cdots, I_{kj}\}$ et un sac à dos de capacité C. Chaque article a un poids w_{ij} et un profit p_{ij}. Le MCKP consiste à sélectionner un item de chaque groupe, à placer dans le sac à dos de sorte que le profit total soit maximisé tandis que le poids total est inférieur à la capacité C du sac à dos.

Nous pouvons réduire le MCKP à notre problème de sélection (PSWPP) en procédant comme suit :

— Chaque groupe d'items G_j est associé à une classe de service S_j ;

— Chaque item I_{ij} est associé à un service candidat s_{ij} ;

— Le profit p_{ij} de chaque item I_{ij} est associé à $(1 - r_{ij})$, où r_{ij} représente le risque de vie privée du service s_{ij} ;

— Le poids w_{ij} de chaque item I_{ij} correspondra à v_{ij}, le nombre de règles non conformes du service s_{ij} ;
— La capacité C du sac à dos correspondra à V, le nombre de règles non conformes toléré par la solution.

Après les transformations précédentes, le MCKP sera représenté comme suit :

$$\begin{cases} \text{maximize } \sum_{i=1}^{n} \sum_{j=1}^{k} s_{ij}(1 - r_{ij}) \\ \text{subject to } \sum_{i=1}^{n} \sum_{j=1}^{k} s_{ij} v_{ij} \leq V \\ \sum_{i=1}^{k} s_{ij} = 1 \, , \; 1 \leq i \leq n \\ s_{ij} \in \{0, 1\} \, , \; 1 \leq i \leq n, \; 1 \leq j \leq k \end{cases} \tag{4.11}$$

La formule : maximize $\sum_{i=1}^{n} \sum_{j=1}^{k} s_{ij}(1 - r_{ij})$ de l'équation 4.11 est équivalent à : minimize $\sum_{i=1}^{n} \sum_{j=1}^{k} s_{ij} r_{ij}$. Si nous réécrivons l'équation 4.11 en utilisant la nouvelle formule équivalente, cette équation sera similaire à l'équation 4.10. Par conséquent, chaque solution à MCKP est également une solution pour PSWPP et vice versa. Comme MCKP est connu pour être un problème NP-hard, alors PSWPP est également NP-hard. □

4.4.3 Représentation du PSWPP sous forme de graphe multi-niveaux

Pour trouver une solution, nous avons représenté le problème PSWPP sous la forme d'un graphe multi-niveaux, contenant un noeud initial (source) et un noeud final (puits ou Sink). Le noeud source représente la requête de l'utilisateur, tandis que le noeud puits est un noeud supplémentaire, ajouté pour être connecté à tous les services du dernier niveau. Dans ce graphe, les noeuds de chaque niveau représentent les services candidat d'une classe donnée, tandis que les arcs représentent le taux de risque de vie privée induit par chaque service. La figure 4.7 donne un exemple de cette représentation. La partie (a) de cette figure montre un simple graphe de vie privée qui représente le flux de données entre les services abstraits, tandis que la partie (b) représente le graphe multi-niveaux correspondant. Selon cette représentation, la question de la recherche d'une solution est équivalente à trouver le chemin le plus court entre le noeud $Requête$ et le noeud $Puits$, de telle sorte que tous les services de ce chemin sont valides. Il est important de noter que la représentation à base de graphe multi-niveaux n'est pas complète. En fait, la recherche d'une solution en utilisant cette représentation se réfère toujours au graphe de vie privée pour vérifier les contraintes de validité.

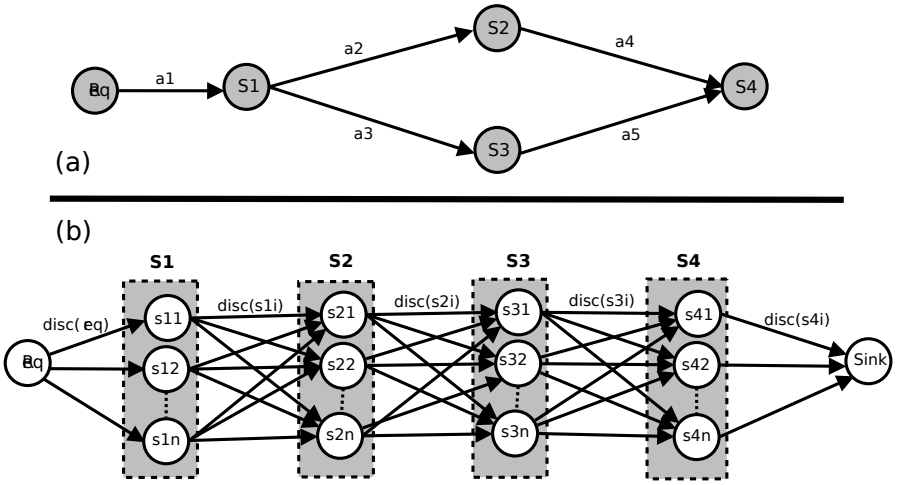


FIGURE 4.7 – Représentation du PSWPP sous forme de graphe multi-niveaux

4.5 Implémentation de la Solution

Cette section présente une vue générale de la solution proposée pour notre problème de sélection. Premièrement, nous discutions la fonction de risque et les motivations derrière le choix de « l'integrale de Choquet » comme fonction de risque. Ensuite, nous décrivons les détails des algorithmes proposés.

4.5.1 L'integrale de Choquet comme Fonction de Risque de Vie Privée.

Pour définir la fonction de risque relative à la vie privée introduite dans la section 4.3.3.9, nous devons utiliser une fonction d'agrégation qui représente le mieux les trois prédicats de vie privée (V, G et T). Dans notre travail, nous avons opté pour une fonction d'agrégation de type *intégrale floue* [Grabisch96], plus précisément une *intégrale de Choquet* [Grabisch96, Marichal00]. Ce choix est justifié par plusieurs raisons : premièrement, une intégrale de Choquet est un opérateur d'agrégation qui généralise plusieurs autres opérateurs classiques, tels que la moyennes arithmétique et ses variantes. Il prend également en compte toutes les interactions et dépendances qui peuvent exister entre les différents critères(prédicats) contrairement aux autres opérateurs d'agrégation classiques. [Marichal00]. Pour utiliser un opérateur d'agrégation de type intégrale de Choquet, il faut d'abord définir la fonction *capacité*(mesure floue) associée.
Les définitions suivantes introduisent les concepts de la fonction capacité ainsi que l'intégrale de Choquet :

Définition 5 (Capacité). *Soit,* $N = \{1, ..., n\}$ *un ensemble de critères. Une mesure floue ou capacité sur N est une fonction* $\mu : 2^N \rightarrow [0, 1]$ *, avec :*

— $\mu(N) = 1, \mu(\emptyset) = 0$ **,**

— *Vérifiant la propriété de monotonie :* $\forall A, B \in 2^N, [A \subseteq B \Rightarrow \mu(A) \leq \mu(B)]$

Définition 6 (L'intégrale de Choquet). *Soit* μ *une capacité sur N. Soit* $a = (a_1, a_2, \cdots, a_n) \in R^+$*. L'intégrale de Choquet de* a *par rapport à* μ *est définie par :*

$$C_\mu(a) = a_{\sigma(1)}\mu(a) + \sum_{i=2}^{n}(a_{\sigma(i)} - a_{\sigma(i-1)})\mu(\{a_{\sigma(i)}, ..., a_{\sigma(n)}\})$$

où σ *est une permutation sur N telle que :* $a_{\sigma(1)} \leqslant a_{\sigma(2)} \leqslant ... \leqslant a_{\sigma(n-1)} \leqslant a_{\sigma(n)}$*. Aussi,* $a_{\sigma(i)} = \{\sigma(i), ..., \sigma(n)\}$*,* $\forall i \in \{1, ..., n\}$*, et* $a_{\sigma(n+1)} = \emptyset$*.*

4.5.1.1 Un Système à base de Logique Floue pour le calcul de la fonction Capacité

Une capacité peut être considérée comme une mesure qui définit l'importance de chaque sous-ensemble de critères (dans notre cas, les prédicats de vie privée) dans un scénario d'agrégation. Son identification est plus complexe qu'une simple attribution de poids à des critères différents. Les auteurs dans [Grabisch08] ont défini de nombreuses méthodes pour identifier la fonction capacité. Toutes ces méthodes supposent soit l'existence d'un ordre entre des groupes de valeurs de critères (défini par un expert), soit une petite base d'apprentissage qui facilite l'identification des poids. Dans notre modèle, l'établissement d'un ordre fondé sur les valeurs des critères de vie privée n'est pas évident, du fait que les critères prennent des valeurs continues dans l'intervalle [0,1]. Au lieu de cela, nous avons utilisé un ensemble de règles floues, qui sont établies par un expert. Les règles floues donnent une estimation floue du risque d'atteinte à la vie privée (élevé, moyen, petit) en fonction des valeurs floues des prédicats V, G et T. En se basant sur ces règles, nous avons avons mis en oeuvre un système flou, qui prend comme entrée un ensemble de valeurs générées aléatoirement des critères de confidentialité V, G et T, et produit comme sortie la valeur de risque de vie privée associée. Cela représente également l'agrégation qu'on souhaite faire avec l'intégrale de Choquet. Les règles suivantes donnent un exemple des règles floues utilisées dans notre système :

— `IF V IS high AND G IS high AND T IS high THEN Risk IS high;`

— `IF V IS high AND G IS high AND T IS low THEN Risk IS medium;`

— `IF G IS low AND V IS low AND (T IS low OR T IS medium) THEN Risk IS low;`

Le système flou nous permet de construire un ensemble de données de la forme : $(V, G, T, Risk)$. Cet ensemble est utilisé par la suite comme une base d'apprentissage pour le calcul de la fonction capacité. Pour le calcul de cette dernière, nous

avons utilisé la méthode des moindres carrées proposée par [Grabisch08].
L'utilisation d'une approche floue pour identifier la fonction de capacité présente plusieurs avantages. Par exemple, un système flou donne une meilleure interprétation qu'un simple ordre fourni par un expert, de plus, l'ensemble des règles floues peut être fourni et révisé par plusieurs experts, aussi, un tel système est plus facile à maintenir et à mettre à jour.
Pour implémenter le système flou, nous avons utilisé la bibliothèque Java jFuzzyLogic [Cingolani12], et pour l'identification de la fonction capacité, nous avons utilisé la boîte à outils Kappalab [Grabisch08], un package du système statistique GNU R [Team14]. La capacité obtenue est décrite dans la table 4.2.

Prédicats	$\{\}$	$\{V\}$	$\{G\}$	$\{T\}$	$\{V,G\}$	$\{V,T\}$	$\{G,T\}$	$\{V,G,T\}$
Capacité	0.0	0.1157	0.251	0.0066	0.5578	0.4356	0.6122	1.0

TABLE 4.2 – Valeurs de la fonction Capacité

Cette table montre les poids des sous-ensembles de prédicats de vie privée. À partir de cette mesure floue, nous pouvons déduire certains indices (par exemple : l'index de Shapley, l'index d'interaction) permettant d'interpréter divers paramètres comme l'importance et les interactions entre les prédicats de vie privée (pour plus de détails, voir [Marichal00]).

4.5.2 Algorithmes proposés

Cette section décrit les algorithmes proposés pour résoudre notre problème de sélection. Nous discuterons d'abord le premier algorithme, qui est une adaptation d'un algorithme de type Best First Search, puis, nous introduisons deux approches déclaratives : l'approche MaxSAT (partial weighted Max-SAT) et l'approche ASP (Answer Set Programming).

4.5.2.1 l'Algorithme Constrained Best First Search (CBFS)

En utilisant la représentation graphique à plusieurs niveaux décrite dans la section 4.4.3, une solution à notre problème consiste à trouver le chemin le plus court entre le noeud "requête"(ou initial) et le noeud "puits" (ou final). Le chemin doit respecter la contrainte de ne contenir que des services valides (voir la section 4.3.3.7). L'approche la plus triviale consiste à adapter un algorithme de recherche de plus court chemin efficace, et qui peut gérer les contraintes de validité. Vu que notre objectif est d'obtenir une solution optimale, des algorithmes non exhaustifs de type meilleur d'abord (Best First Search :BFS) comme A * [Hart68] et ses va-

riantes [Goldenberg14], qui élaguent stratégiquement l'espace de recherche grâce à des heuristiques, sont de bons candidats. En général, un algorithme BFS maintient deux listes de noeuds, une liste Open et une liste Closed. La liste Closed est utilisée pour stocker les noeuds déjà explorés, tandis que la liste Open est une file d'attente avec priorité qui ordonne les noeuds non-explorés en utilisant une fonction de coût $f(s) = g(s) + h(s)$. Plus le coût $f(s)$ d'un noeud s est faible, plus sa priorité est élevée. Dans cette fonction de coût, $g(s)$ représente le coût du passage du noeud initial au noeud s, et $h(s)$ représente une heuristique d'estimation du coût pour atteindre le noeud final à partir du noeud s. Une heuristique admissible (c'est-à-dire ne jamais surestimer le coût réel) offre une garantie d'optimalité pour ce type d'algorithmes. Dans notre travail, nous avons utilisé le même type d'algorithme BFS, avec une adaptation pour gérer les contraintes de validité. Pour obtenir une heuristique admissible, nous avons relaxé (assouplir) les contraintes de validité de notre problème. La relaxation est faite de sorte que la valeur heuristique d'un noeud s_{ij} (qui représente le service j de la classe i) est égale au chemin le plus court entre ce noeud et le noeud final, sans tenir compte des contraintes de validité. En d'autres termes, la composition représentée par le chemin allant du noeud s_{ij} au noeud final peut être invalide (contient des services invalides). Notons que cette heuristique est facile à calculer si les services de chaque classe sont triés par ordre croissant de leurs valeurs de risque de vie privée. Dans ce cas, l'heuristique du noeud s_{ij} est calculée comme suit :

$$h(s_{ij}) = risk(s_{ij}) + \sum_{k=i+1}^{n} risk(s_{1k}) \tag{4.12}$$

Où n est le nombre de classes. Par exemple, dans la représentation de la figure 4.7 (b), si nous supposons que les services de toutes les classes sont triés, l'heuristique du service s_{22} est : $h(s_{22}) = risk(s_{22}) + risk(s_{13}) + risk(s_{14})$.

Le listing 2 décrit les principales étapes de notre algorithme. Cet algorithme prend comme entrée une représentation graphique multi-niveaux sous forme de la structure de données ($Base(S, PP, PR, PR, Req)$) et retourne une composition privée optimale ($OptComp$) si elle existe. Dans cet algorithme, nous supposons que le risque relatif à la vie privée de chaque service est déjà calculé et que les services de chaque classe sont triés par ordre croissant de leur valeur de risque de vie privée. L'algorithme calcule d'abord la valeur heuristique pour chaque service (lignes 1-8). Ensuite, il lance la procédure de recherche en insérant le noeud $requête$ dans la file d'attente prioritaire (ligne 9). Les lignes de 10 à 22 représentent les étapes de la boucle principale de la procédure de recherche. Cette procédure vise à atteindre à chaque fois le noeud qui porte le coût $f(s)$ le plus bas (ligne 11). Si ce noeud ($CurrentService$) est un noeud $final$, alors l'algorithme renvoie le chemin correspon-

Algorithme 2 : l'Algorithme Constrained Best First Search

Input : $Base(S, PP, PR, Req)$,
$S = \{S_i\}$, $|S|$ = number of classes ;
$S_i = \{s_{ij}\}$: Set of candidate services of class S_i ;
$PP = \{PP_{s_{ij}}\}$:Set of privacy policies of all services ;
$PR = \{PR_{s_{ij}}\}$:Set of privacy requirements of all services ;
$Req = (PR_{req}, PP_{req})$: Policies and requirements of user request ;
Output : Optimal composition : $OptComp$

```
/* le calcul de la valeur heuristique pour chaque service.        */
```
1 **for** $i \leftarrow 1$ **to** $|S|$ **do**
2 $\quad S_i \leftarrow S[i]$
3 $\quad$ **for** $j \leftarrow 1$ **to** $|S_i|$ **do**
4 $\quad\quad s_{ij} \leftarrow S_i[j]$
5 $\quad\quad h \leftarrow calculateHeuristic(s_{ij})$
6 $\quad\quad setHeuristic(h, s_{ij})$
7 $\quad$ **end**
8 **end**
```
/* Recherche de la solution.                                      */
```
9 $OpenList.Add(Req)$; // La file d'attente prioritaire.
10 **while** $|OpenList| > 0$ **do**
11 $\quad CurrentService \leftarrow OpenLis.firstElement()$
12 $\quad$ **if** *isGoal*$(CurrentService)$ **then**
13 $\quad\quad$ **return** $OptComp \leftarrow path(CurrentService)$
14 $\quad$ **end**
15 $\quad OpenList.remove(CurrentService)$
16 $\quad NextLevelList \leftarrow generateNext(CurrentService)$
17 $\quad$ **foreach** s_i *in* $NextLevelList$ **do**
18 $\quad\quad DistanceFromStart \leftarrow$
$\quad\quad getRisk(CurrentService) + getDistanceFromStart(CurrentService)$
19 $\quad\quad setDistanceFromStart(DistanceFromStart, S_i)$
20 $\quad\quad OpenList.Add(s_i)$
21 $\quad$ **end**
22 **end**
23 **return** *solution does not exist*

dant (lignes 12-14). Si ce n'est pas le cas, le noeud $CurrentService$ sera supprimé de la file d'attente (ligne 15) et l'algorithme génère le niveau suivant des services correspondants (voisins de $CurrentService$) en utilisant la fonction $generateNext()$ (ligne 16). Cette fonction de génération qui fait la différence entre l'algorithme proposé et les autres algorithmes BFS. En effet, notre algorithme génère uniquement les services compatibles avec les services du chemin allant du noeud *requête* jusqu'au noeud $CurrentService$. Cela signifie un nombre plus réduit de noeuds générés, ce qui améliore les performances. Par la suite, pour chaque service généré s', l'algorithme calcule la distance qui sépare s' et le noeud *requête*, puis il ajoute s' à la file d'attente (lignes 17-21). L'algorithme continue jusqu' à ce qu'il trouve une solution ou que la file d'attente soit vide.

4.5.2.2 L'approche Max-SAT

Cette approche consiste à coder le problème PSWPP en tant qu'instance *Max-SAT partielle et pondérée* (partial weighted Max-SAT :PWMSAT). Ensuite, un solveur Max-SAT est appelé pour trouver la solution. Une instance PWMSAT contient deux formules CNF pondérées. Les deux formules représentent les clauses "**hard**" et "**soft**". L'objectif est de trouver une affectation de variables qui satisfait toutes les clauses "hard" tout en maximisant le poids total des clauses "soft" satisfaites. Pour encoder notre problème de sélection en tant que PWMSAT, nous avons procédé comme suit :

1. le fait de ne sélectionner qu'un seul service s_{ij} de chaque classe S_i est représenté par le type spécial de contraintes de cardinalité, souvent appelé *contrainte exactement-une* et représentée par $\sum_{j=1}^{|S_i|} x_{ij} = 1$. Pour encoder cette contrainte, nous avons utilisé l'encodage *Commander-Variable Encoding* proposé par Klieber et Kwon [Klieber07]. Cet encodage consiste à diviser les variables de chaque classe abstraite en m groupes disjoints : $G_1 \dots G_m$. Dans chaque groupe, une variable de commande c_i est introduite. La variable de commande doit être vraie si (au moins) l'une des variables de son groupe est vraie ; sinon, elle doit être fausse. Ainsi, pour chaque classe de services S_i ($|S_i| = n$), nous avons les formules CNF suivantes :
   — Au plus une variable d'un groupe G_k peut être vraie :

$$\bigwedge_{x_{ij} \in G_k} \bigwedge_{x_{ij'} \in G_k, j' < j} \neg x_{ij} \vee x_{ij'} \tag{4.13}$$

   — Si la variable de commande d'un groupe G_k est vraie, alors au moins une des variables du groupe doit être vraie :

$$\neg c_k \vee \bigwedge_{x_{ij} \in G_k} x_{ij} \tag{4.14}$$

   — Si la variable de commande d'un groupe G_k est fausse, alors aucune des variables du groupe ne peut être vraie :

$$\bigwedge_{x_{ij} \in G_k} (c_k \vee \neg x_{ij}) \tag{4.15}$$

   — Exactement une des variables de commande est vraie, ce qui est encodée par une application récursive de la méthode *Commander-Variable Encoding*.
2. la contrainte de validité qui exige que tous les services d'une composition donnée doivent être valides (voir l'équation 4.7) est encodée comme suit : $\forall S_i, S_{i'} \in S$, pour tous les services $s_{ij} \in S_i$ et les services $s_{i'j'} \in S_{i'}$, Si :

$(s_{i\prime j\prime} \in PRD^{s_{ij}} \wedge Nconf(s_{i\prime j\prime}, s_{ij}) = 1)$ (voir les équations 4.6 et 4.7) nous ajoutons la contrainte spécifiant que les services s_{ij} et $s_{i\prime j\prime}$ ne peuvent pas être dans la même composition. Cette contrainte est exprimée par $(x_{ij} \Rightarrow \neg x_{i\prime j\prime} \wedge x_{i\prime j\prime} \Rightarrow \neg x_{ij})$ ce qui est équivalent à la clause :

$$\neg x_{ij} \vee \neg x_{i\prime j\prime} \tag{4.16}$$

3. les clauses "hard" sont représentées par les clauses utilisées pour encoder la contrainte de cardinalité ($\sum_{j=1}^{|S_i|} x_{ij} = 1$) et la contrainte de validité (formula 4.16).
4. les clauses "soft" sont représentées par les formules atomiques x_{ij} pondérées par la valeur $(1 - risk(s_{ij}))$. La maximisation de la somme des poids $(1 - risk(s_{ij}))$ est équivalente à minimiser la somme des valeurs $risk(s_{ij})$, par conséquent, la solution correspondante aura un risque de vie privée minimum.

Pour analyser les performances de notre encodage en termes de nombre de variables et de clauses produites, nous considérons que nous avons k classes abstraites et n services dans chaque classe.

1. ***Nombre de variables*** : Le fait que chaque service est représenté par une variable booléenne nécessite des variables $n * k$. De plus, l'encodage de la contrainte de cardinalité avec la méthode *Commander-Variable Encoding* produit $\frac{n}{2}$ variables supplémentaires pour chaque classe (voir [Klieber07]). Ainsi, un total de $k * \frac{n}{2}$ variables supplémentaires. Par conséquent, le nombre total de variables requises pour encoder une instance du problème PSWPP est :

$$\frac{3}{2} * n * k$$

.

2. ***Nombre de clauses :*** Le nombre total de clauses produites par l'encodage proposé est égal au nombre de clauses hard plus le nombre de clauses soft. Pour les clauses hard , l'encodage *Commander-Variable Encoding* de la contrainte de cardinalité (avec une taille de groupe fixée à 3) produit $\frac{7}{2} * n$ clauses par classe (voir [Klieber07]). Ainsi, un total de $\frac{7}{2} * n * k$ clauses. D'autre part, dans le pire des cas, l'encodage de la contrainte de validité produit environ $n^2 * (k - 1)$ clauses binaires pour chaque classe (si tous les services ne sont pas conformes). Donc, un nombre total de clauses $n^2 * (k - 1) * k$. L'encodage des clauses soft nécessite $n * k$ clauses unaires. Par conséquent, le nombre total de clauses utilisées pour encoder une instance du problème

PSWPP est :

$$\frac{7}{2} * n * k + n^2 * (k-1) * k + n * k.$$
$$= n^2 * (k^2 - k) + \frac{9}{2} * n * k.$$

4.5.2.3 l'Approche ASP

Cette approche consiste à encoder le PSWPP en tant que programme ASP, puis un solveur ASP sera utilisé pour trouver la solution. En général, un encodage ASP se compose de deux parties principales : une partie *Génération* (ou estimation) et une partie *Test* (ou contrôle) [Lifschitz02]. La partie *Génération* définit des règles et des faits qui génèrent des candidats potentiels de modèles stables, généralement par le biais de constructions non déterministes, alors que la partie *Test* correspond à la définition des contraintes du problème, et élimine également les candidats invalides. D'autres parties d'encodage peuvent également exister, comme la partie *Définition* qui définit des prédicats auxiliaires, ou la partie *Optimisation* dans le cas d'un problème d'optimisation. Pour encoder notre problème, nous avons défini les parties suivantes :

1. **La partie Génération** : cette partie consiste en un ensemble de prédicats qui génèrent tous les services possibles de l'instance de problème. Les prédicats utilisés sont :
   ```
   class(1..m).
   serviceNb(0..n).
   {serviceId(I,J)} :- class(J),serviceNb(I).
   ```
   Les prédicats `class/1` et `serviceNb/1` définissent respectivement le nombre de classes et le nombre de services par classe de l'instance du problème. La règle avec la tête `{serviceId(I,J)}` génère l'ensemble de tous les `serviceId(I,J)` possibles (l'ensemble des réponses). Ce prédicat (c.-à-d. `serviceId(I,J)`) représente le service `J` de la classe `I`.
2. **La partie définition** : Cette partie définit tous les prédicats auxiliaires utilisés dans la définition de l'instance de problème. Les prédicats utilisés sont spécifiés ci-dessous :
   ```
   risk(I,J,R).
   pp(I,J,Att,Pr,V,G,T).
   pr(I,J,Att,Pr,V,G,T).
   depend(I,K).
   conforme(S1,I,S2,K):-serviceId(S1,I),serviceId(S2,K),not depend(I,K).
   conforme(S1,I,S2,K):-serviceId(S1,I),serviceId(S2,K),depend(I,K),
                        pr(S1,I,A,Pr1,V1,G1,T1),pp(S2,K,A,Pr2,V2,G2,T1),
                        Pr1=Pr2,V1 >= V2 , G1 >= G2,T >= T1.
   ```
   Le prédicat `risk/3` est utilisé pour définir le risque de vie privée `R` relatif au service `J` de la classe `I`. Les deux prédicats `pp/7` et `pr/7` défi-

nissent respectivement les politiques et les exigences de vie privée du service `J` de la classe `I`. Ainsi, le prédicat `pp(I,J,Att,Pr, V,G,T).` (resp. `pr(I,J,Att,Pr,V,G,T).`) définit les valeurs des dimensions de vie privée : objectif (`Pr`), Visibilité (`V`), Granularité (`G`) et Temps de rétention (`T`) relatifs à l'attribut privé `Att`. Le prédicat `depend/3` spécifie s'il existe une relation entre les deux classes `I` et `K`. Ce prédicat est ensuite utilisé dans le prédicat `conforme/4` pour implémenter la règle de conformité entre deux services. Ainsi, deux services `S1` (de la classe `I`) et `S2` (de la classe `K`) sont conformes si le prédicat `conforme/4` est vrai. Ce prédicat est vrai si l'une des deux situations est vraie : a) si aucune relation n'existe entre les services (ou classes) abstraits dans lesquels les deux services appartiennent, b) si les deux services relatifs sont conformes selon l'équation 4.5 du modèle de vie privée.

3. **La partie Test** : cette partie est constituée de règles représentant les contraintes de notre problème. Dans notre encodage, nous avons utilisé les deux contraintes suivantes :
   ```
   :- I= 0..m, not 1 { serviceId(I,J)} 1.
   :- serviceId(I,S1),serviceId(K,S2), not conforme(S1,I,S2,K).
   ```
   La première contrainte stipule qu'un modèle stable ne peut pas contenir plus d'un fait de type `serviceId(I, _)` pour une classe donnée `I`. En d'autres termes, cette règle garantit que la solution ne doit contenir qu'un seul service de chaque classe. La deuxième contrainte indique que si deux services ne sont pas conformes, ils ne peuvent pas être dans le même modèle stable.
4. **La partie Optimisation** : Cette partie est une directive qui indique au solveur ASP de calculer le modèle stable optimal. Dans notre encodage, nous utilisons l'instruction suivante :
   ```
   #minimize { R,I,J : serviceId(I,J),risk(I,J,R) }.
   ```
   cette instruction minimise la somme des valeurs de risque de vie privée associées à chaque service des modèles stables.

4.6 Evaluation

D'un point de vue de protection de la vie privée, tous les algorithmes proposés fournissent (par conception) la meilleure composition qui satisfasse toutes les contraintes de vie privée des utilisateurs et les fournisseurs de services. Ainsi, notre évaluation affecte principalement l'efficacité et la scalabilité de chaque algorithme.

4.6.1 Description de l'ensemble de données de l'évaluation

Pour évaluer les approches proposées, nous avons utilisé un ensemble de données généré aléatoirement, ceci est dû principalement au manque des bases ap-

propriées [1]. Le processus de génération est basé sur plusieurs travaux [Feng15, Huang12, Cherifi10, Kil09, Oh08]. Ces derniers affirment que la majorité des réseaux de services Web ont les caractéristiques des réseaux *small-world* [Watts98] ou des réseaux scale-free [Bollobás03]. Par conséquent, nous avons généré deux types de bases : une base small-world et une base scale-free. Les graphes de vie privée de ces ensembles de données ont été générés en utilisant l'outil Networkx [Hagberg13], dans lequel le modèle small-word est basé sur les graphes de Watts-Strogatz [Watts98], tandis que le modèle scale-free est basé sur l'algorithme de Bollobas et al. [Bollobás03]. La table 4.3 décrit les deux ensembles de données générés en termes de taille de composition et le nombre total de règles de vie privée (règles de politique de vie privée (PP), et règles d'exgigence de vie privée (PR)).

	La base Small-world			**La base Scale-free**		
Taille de comp	**PP**	**PR**	**Total**	**PP**	**PR**	**Total**
3	48	36	**84**	24	24	**48**
4	72	48	**120**	48	36	**84**
5	48	41	**89**	72	36	**108**
6	72	49	**121**	120	48	**168**
7	96	64	**160**	144	72	**216**
8	72	64	**136**	120	100	**220**
9	144	107	**251**	117	93	**210**
10	144	98	**242**	144	91	**235**

TABLE 4.3 – Description des ensembles de données générés.

Toutes les expérimentations sont réalisées sur un processeur Intel Core i7-4700HQ 2.40GHz, dans une machine avec une mémoire de 6 Go, exécutant une distribution Ubuntu 64 bits. L'Algorithme Constrained Best First Search (CBFS) est implémenté avec Java 8 sous l'environnement NetBeans. Nous avons utilisé le solveur QMaxSAT [Koshimura12] pour implémenter l'approche Max-SAT, et le solveur Clingo [Gebser14] pour implémenter l'approche ASP. Notons que pour l'approche Max-SAT, nous avons essayé une version de MaxSatz [Li09b] appelée Maxsatz2013f et aussi le solveur CCLS_to_Akmaxsat [2] [Luo15], les résultats n'étaient pas aussi bons que ceux de QmaxSAT. Tous les ensembles de données et les implémentations des algorithmes utilisés peuvent être trouvés sur le lien suivant [3].

1. Les bases disponibles, comme `http://www.uoguelph.ca/~qmahmoud/qws/`, ne sont adaptés que pour le problème de composition à base de QoS
2. `http://maxsat.ia.udl.cat/solvers/5/CCLS_to_akmaxsat_binaries.zip-201604080802`
3. `https://www.dropbox.com/s/ag8w12tvonulib2/PrivacyDatasets.tar.gz?dl=0`.

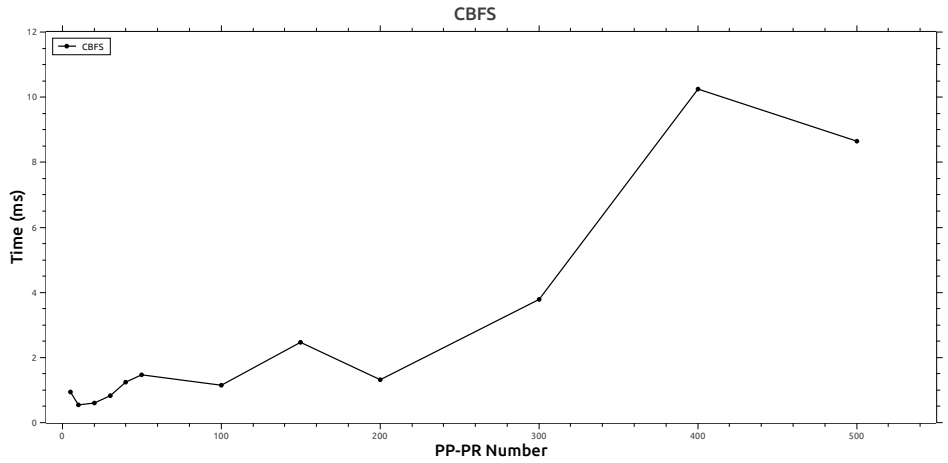
4.6.2 Influence du nombre de règles de vie privée (PP-PR) sur le temps d'exécution

Cette expérimentation évalue l'influence du nombre de règles de vie privée (PP-PR) sur l'efficacité (en termes de temps d'exécution) des algorithmes proposés. Pour cela, nous avons fixé la taille de la composition à 2 et le nombre de services dans chaque classe à 50, tout en faisant varier le nombre de règles de vie privée de chaque service de 5 à 500. Les résultats de cette expérimentation sont présentés dans la Table 4.4 et la Figure 4.8.

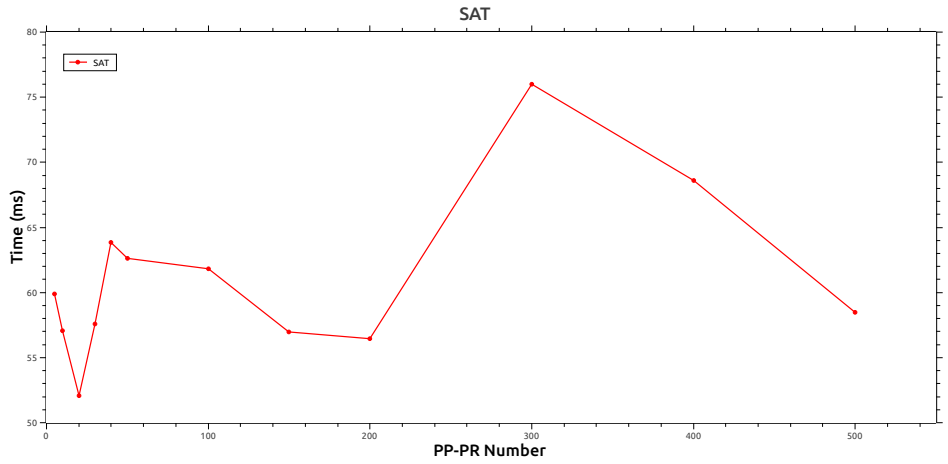
	Algorithmes (ms)		
Nombre PP-PR	**CBFS**	**SAT**	**ASP**
5	0.94	59.9	34.72
10	0.55	57.08	41.64
20	0.6	52.08	54.98
30	0.82	57.6	67.7
40	1.25	63.86	82.72
50	1.47	62.62	101.56
100	1.14	61.8	174.26
150	2.46	56.98	264.58
200	1.32	56.44	320.26
300	3.78	75.98	580.34
400	10.25	68.6	775.08
500	8.64	58.48	908.48

TABLE 4.4 – Influence du nombre de règles(PP-PR) sur l'efficacité des algorithmes proposés.

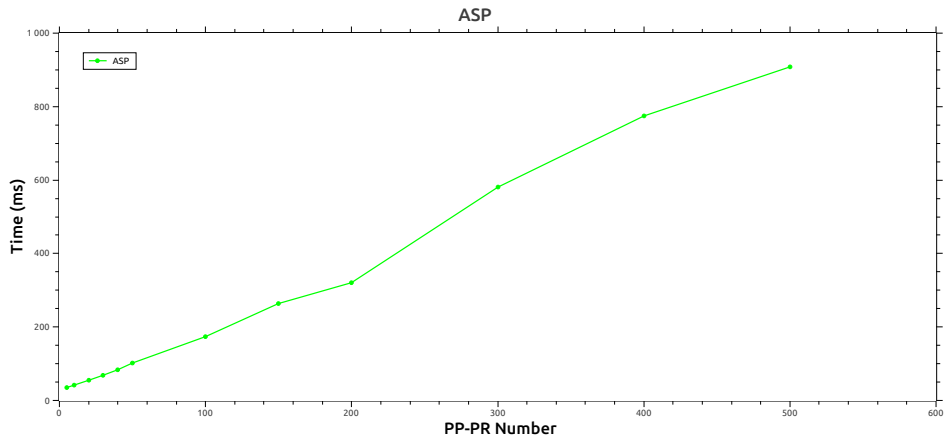
Les résultats de cette expérimentation montrent que pour les algorithmes CBFS et ASP, la complexité des interactions entre les services (exprimée par le nombre de règles de vie privée) a une influence apparente sur le temps d'exécution. Généralement, le temps d'exécution augmente avec l'augmentation du nombre de règles de vie privée. Il existe quelques exceptions pour l'algorithme CBFS, où le temps d'exécution est plus court malgré un plus grand nombre de règles (par exemple le temps d'exécution pour 150 règles est moins que celui de 200 règles, la même remarque pour 400 et 500). Ceci s'explique par le fait que le temps d'exécution de l'algorithme CBFS est déterminé par deux facteurs (voir Algorithme2) : le premier, est le temps nécessaire pour vérifier la conformité des services (qui est fortement influencé par le nombre de règles de vie privée), le second, est le temps nécessaire pour la manipulation de la file d'attente prioritaire (insertion, suppression et tri). Le temps d'exécution du second facteur est influencé par la taille de la file d'attente prioritaire, qui peut augmenter ou diminuer selon la requête de l'utilisateur et indépendamment du nombre de règles de vie privée. Dans certaines situations, il peut y avoir une plus grande file d'attente avec un plus petit nombre de règles de



(a)



(b)



(c)

FIGURE 4.8 – Influence du nombre de règles (PP-PR) sur l'efficacité des algorithmes proposés : (a) CBFS ,(b) Max-SAT, (c) ASP.

vie privée. Cela entraîne un temps plus grand pour le traitement des files d'attente, ce qui augmente considérablement le temps d'exécution.
Les résultats sont différents pour l'approche Max-SAT, dans laquelle on ne peut pas voir une réelle dépendance entre le temps d'exécution et le nombre de règles de vie privée. Ceci est principalement dû à notre encodage SAT qui se fait au niveau services. Dans cet encodage, seuls les services incompatibles qui sont encodés en tant que contraintes (voir équation 4.16), et non pas les détails des interactions. Cela rend notre encodage indépendant de la complexité des interactions, donc un temps d'exécution indépendant aussi.
Si nous comparons l'efficacité de chaque algorithme pour cette expérimentation (voir Figure 4.9), nous pouvons voir clairement que les approches CBFS et Max-SAT fournissent de bien meilleurs résultats que l'approche ASP, avec un léger avantage pour l'algorithme CBFS par rapport à l'approche Max-SAT.

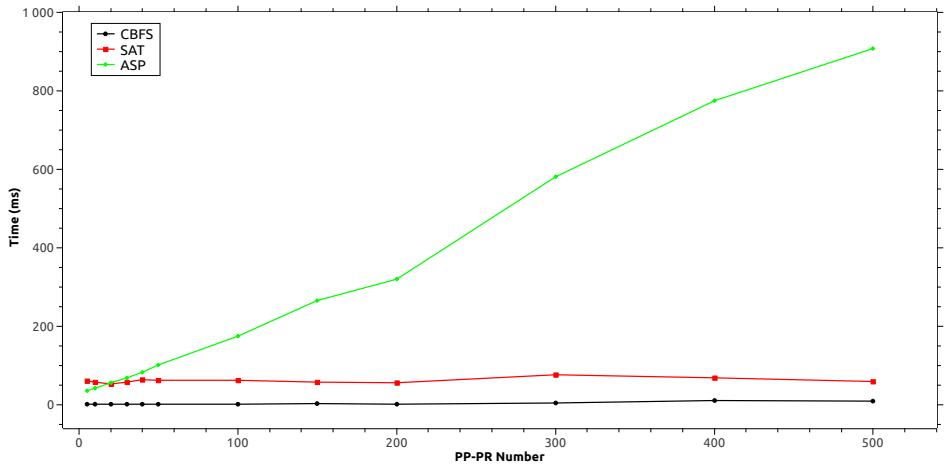


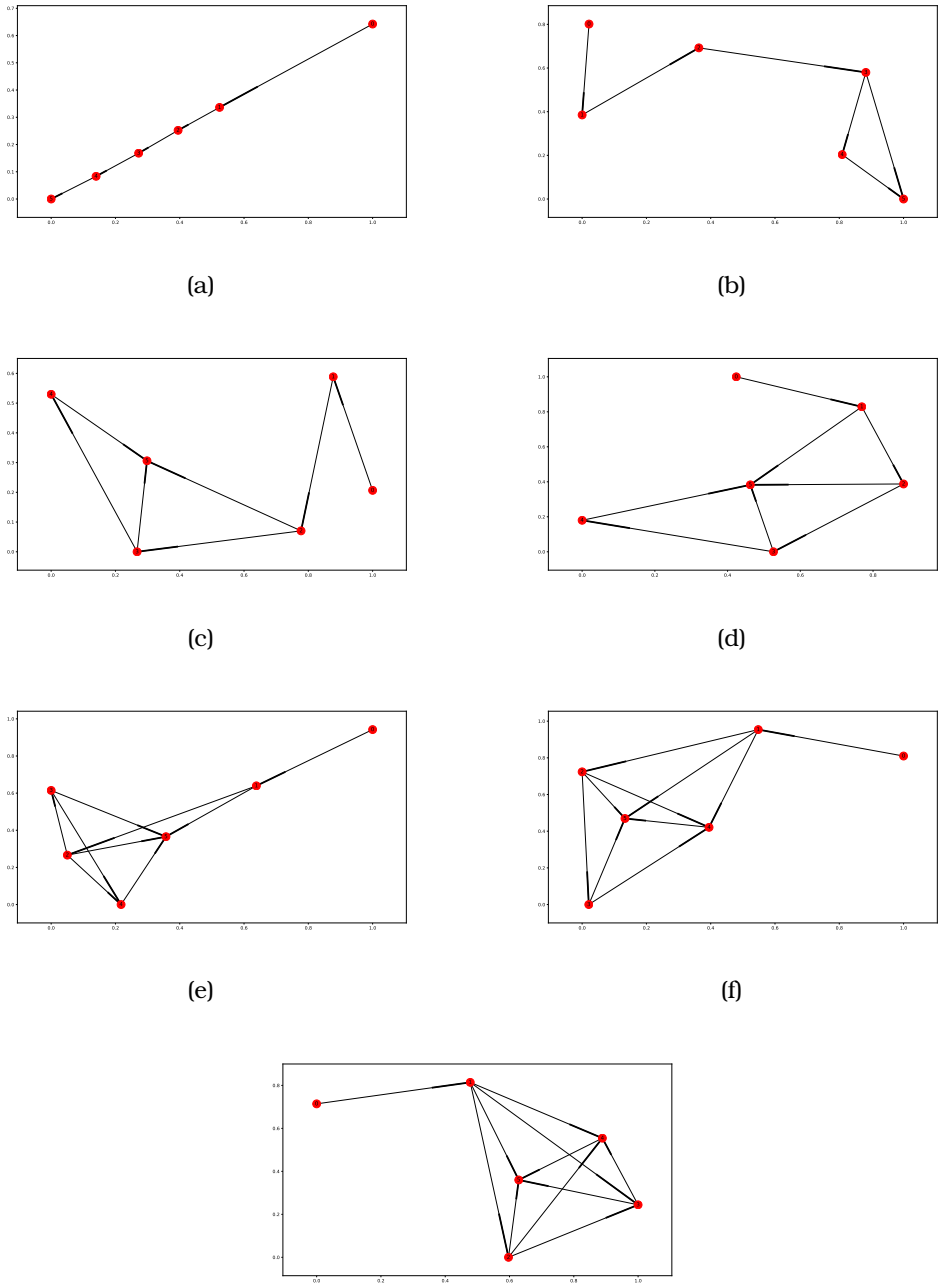
FIGURE 4.9 – Comparaison de l'efficacité des algorithmes pour l'influence du nombre de règles (PP-PR) sur le temps d'exécution.

4.6.3 Influence de la complexité des interactions des services abstraits sur l'efficacité des approches proposées

Le but de cette expérimentation est d'étudier comment la complexité des dépendances entre les services abstraits affecte la performance des algorithmes proposés. Pour ce faire, nous avons commencé avec une composition abstraite séquentielle de cinq services, après quoi nous ajoutons un arc (une dépendance) entre les services à la fois. Par conséquent, nous avons obtenu 7 graphes de vie privée avec un nombre croissant d'interactions allant de 5 à 11 (voir Figure[4] 4.10). L'utilisation d'une composition ayant seulement cinq services est basée sur le travail de [Huang12] qui a étudié une collection de 6092 compositions et a montré que $95,45\%$

4. Le noeud 0 dans tous les graphes représente la requête de l'utilisateur

d'entre elles ne contenaient pas plus de 5 services.
Notons que dans cette expérimentation, un arc d'un graphe de vie privée représente un seul attribut de données, et que chaque service abstrait compte 50 services candidats.



(a) (b) (c) (d) (e) (f) (g)

FIGURE 4.10 – Graphes de vie privée utilisés pour tester l'influence de la complexité des interactions entre les services sur l'efficacité des approches proposées.

Les résultats de cette expérimentation sont présentés dans le tableau 4.5 et la figure 4.11.

	Algorithmes		
Nombre d'interactions	**CBFS (ms)**	**SAT (ms)**	**ASP (ms)**
5	2.41	663.89	279.71
6	3.93	605.82	292.12
7	25.08	464.57	283.47
8	36.34	331.8	284.07
9	28.65	478.01	269.77
10	10.87	340.59	290.07
11	7.67	263.14	257.63

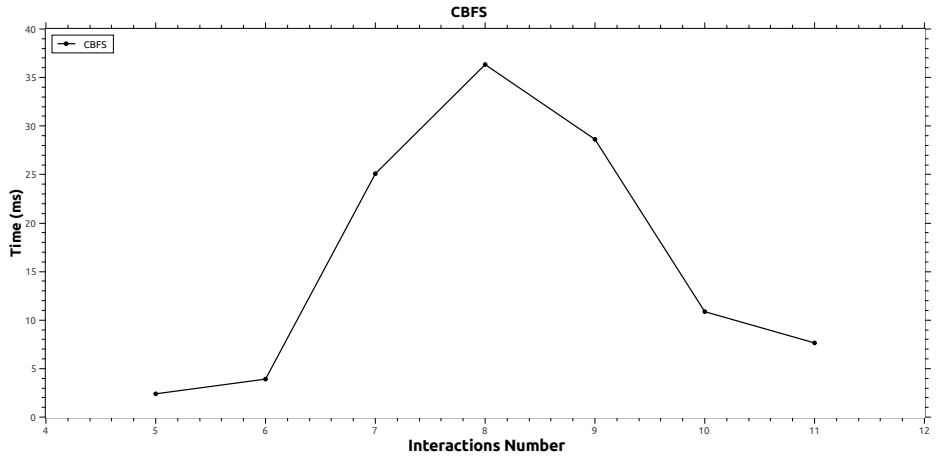
TABLE 4.5 – L'influence de la complexité des interactions de services sur l'efficacité des algorithmes proposés.

Les résultats n'ont pas montré une dépendance ou un modèle clair entre la complexité des interactions et le temps de calcul. Nous pouvons voir pour tous les algorithmes qu'il existe pour certains graphes, un temps de calcul plus court pour des interactions plus grandes et vice versa.
Si nous comparons l'efficacité de chaque algorithme pour cette expérimentation (Figure 4.12), nous voyons clairement que l'algorithme CBFS offre de meilleures performances, tandis que l'approche Max-SAT présente la moindre performance.

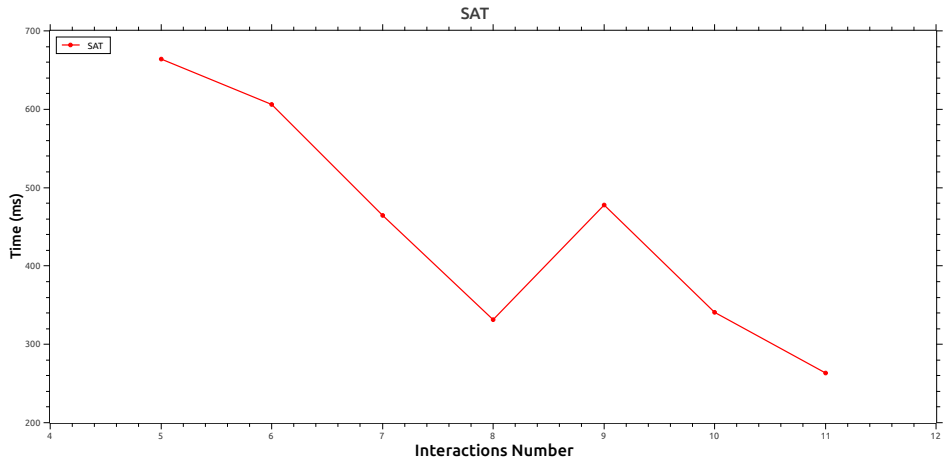
4.6.4 Influence de la taille de la composition sur l'efficacité des approches proposées

Cette expérimentation évalue l'influence de la taille de la composition sur l'efficacité des approches proposées. Pour ce faire, nous avons fait varier la taille de la composition de 3 à 10, tandis que nous avons fixé à 50 le nombre de services candidats de chaque classe. Les résultats de cette expérimentation sont présentés dans Table 4.6 et figure 4.13.

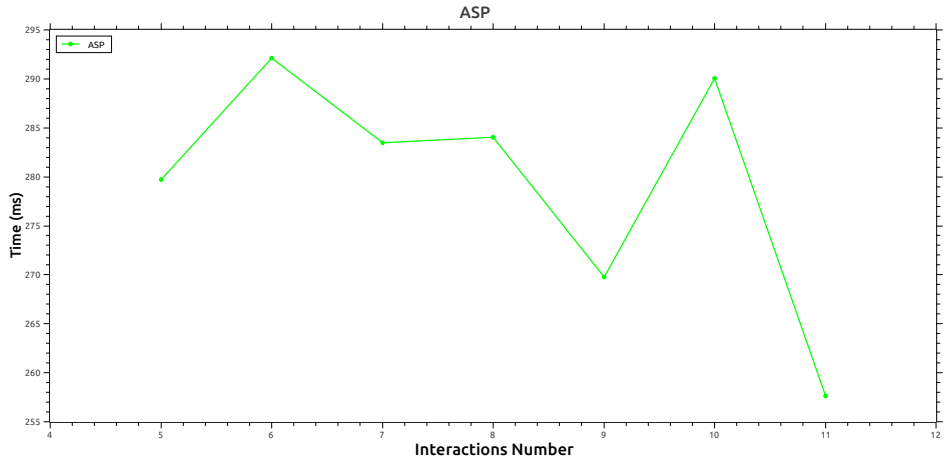
La première observation tirée de ces résultats, est que le temps de calcul augmente avec la taille de la composition. Cela est valable pour tous les ensembles de données (Small-word et Scale-free) et algorithmes, à quelques exceptions près pour l'algorithme CBFS. L'augmentation est plus significative pour l'algorithme CBFS, de sorte que nous avons obtenu un *timeout* pour les compositions 10 de l'ensemble de données Small-world et 9 et 10 de l'ensemble de données Scale-free (voir Table 4.6). Pour l'algorithme CBFS, on peut justifier cette augmentation par le fait qu'une augmentation de la taille de la composition implique une augmentation de la profondeur de l'état final de l'arbre de recherche (le noeud puits dans notre représentation multi-niveaux du problème (voir Figure 4.7)). Par conséquent, un temps de



(a)



(b)



(c)

FIGURE 4.11 – L'influence de la complexité des interactions de services sur l'efficacité des algorithmes proposés : (a) CBFS ,(b) Max-SAT, (c) ASP.

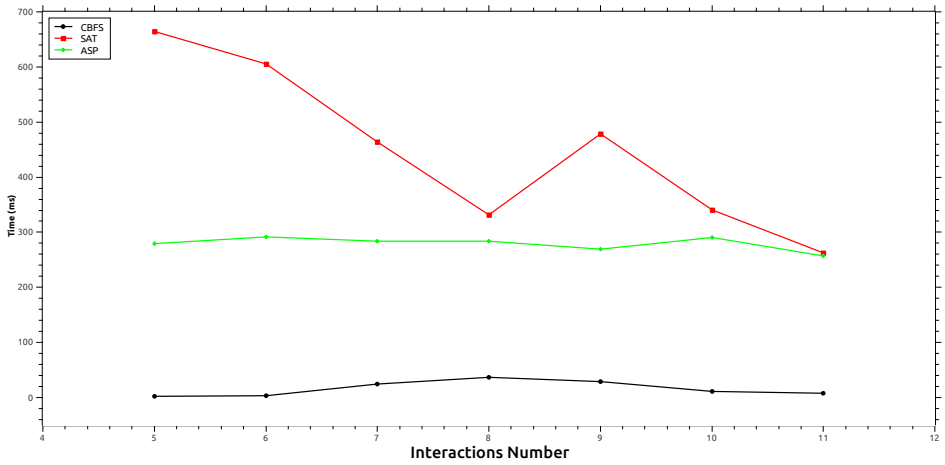


FIGURE 4.12 – Comparaison de l'efficacité des algorithmes proposés pour l'influence de la complexité des interactions de services sur le temps d'exécution.

	Small-world Dataset			**Scale-free Dataset**		
Compo size	**CBFS**	**SAT**	**ASP**	**CBFS**	**SAT**	**ASP**
3	5.58	42,46	63,38	5,99	42,46	63,38
4	8.83	114,87	124,42	23,36	114,87	124,42
5	1 400.84	367,11	190,21	77,37	367,11	190,21
6	1 120.19	287,68	291,01	32,55	287,68	291,01
7	685.97	1161,54	498,01	334,5	1161,54	498,01
8	52 404	1948,77	642,58	170885	1948,77	642,58
9	1 571.62	1315,63	1013,18	timeout	1315,63	1013,18
10	timeout	2980,28	1560,82	timeout	2980,28	1560,82

TABLE 4.6 – L'influence de la taille de la composition sur l'efficacité des algorithmes proposés.

calcul plus important. En ce qui concerne les approches déclaratives (Max-SAT et ASP), une augmentation de la taille de la composition implique une augmentation du nombre de variables et de clauses des formules booléennes générées, donc un temps de calcul plus important.

Un pic anormalement élevé est observé pour l'algorithme CBFS au niveau de la composition de taille 8 de la base Small-world. Cela fait que le temps de calcul de la composition 9 est beaucoup plus faible que celui de la composition 8. Cela nous amène à dire que le fait qu'une augmentation de la taille de la composition implique une augmentation du temps de calcul ne peut pas être considéré comme une règle générale. En fait, l'heuristique utilisée et la spécificité de chaque instance duproblème peuvent avoir un impact dramatique sur le temps de calcul. Dans notre cas, pour trouver une solution, l'algorithme CBFS gère une file d'attente prioritaire d'une taille moyenne de 400 dans la composition 9, alors que la taille moyenne de la file d'attente prioritaire dans la composition 8 est d'environ 4360 (10 fois plus grande), ce qui peut expliquer le pic de temps d'exécution observé.

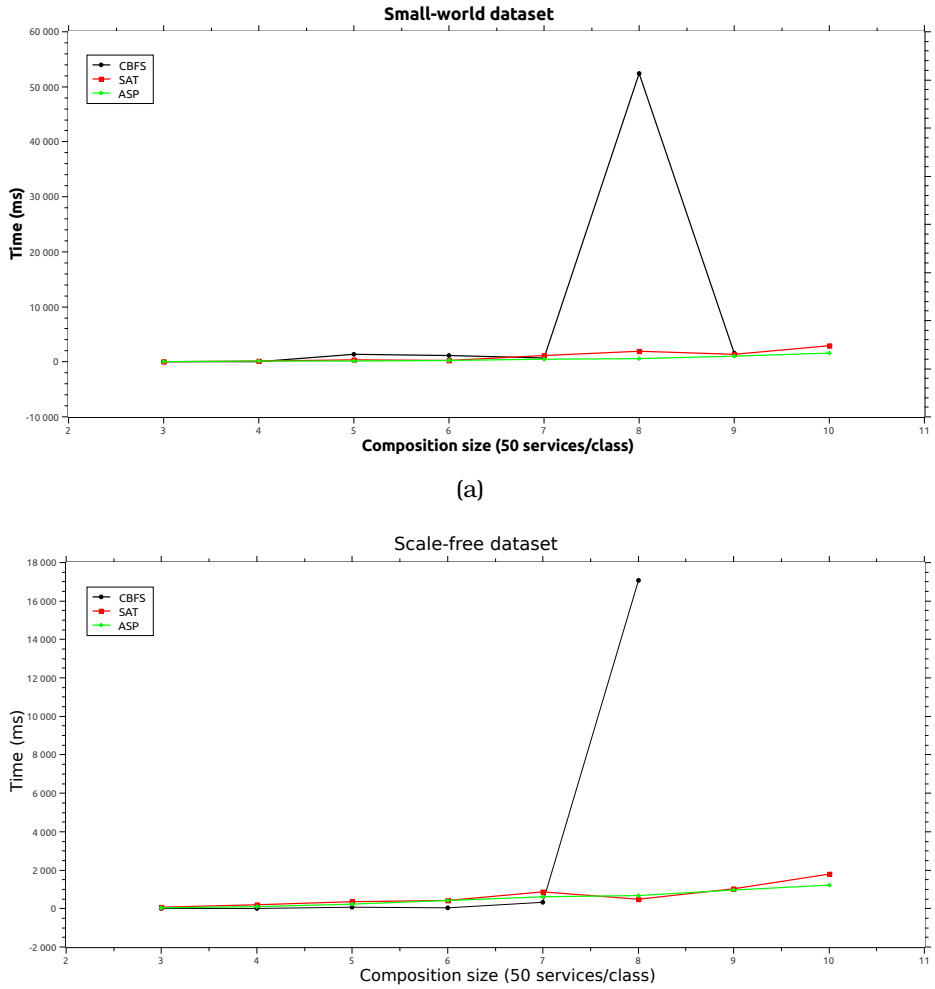


FIGURE 4.13 – L'influence de la taille de la composition sur l'efficacité des algorithmes proposés : (a) Small-world dataset, (b) Scale-free dataset.

Nous remarquons également que l'efficacité de l'algorithme CBFS est meilleure pour les petites compositions (compositions 3 et 4 de la base Small-world, et les compositions de 3 à 7 de la base Scale-free). Cependant, à mesure que la taille augmente, les approches déclaratives prennent l'avantage, et la différence devient très importante.

4.6.5 Influence du nombre de services candidats par classe sur l'efficacité des approches proposées

Notre dernière expérimentation évalue l'influence du nombre de services candidats sur le temps de calcul des approches proposées. Pour ce faire, nous avons fixé la taille de la composition à 4, tout en faisant varier le nombre de services par classe de 50 à 1000. Les résultats de cette expérimentation sont présentés dans la Table 4.7 et la Figure 4.14.

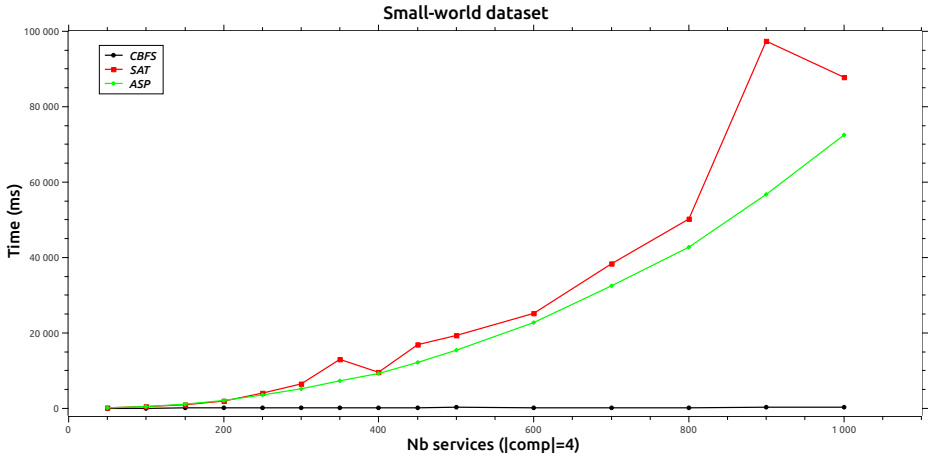
Les résultats de cette expérimentation montrent clairement que le temps de cal-

	Small-world Dataset			**Scale-free Dataset**		
Compo size	**CBFS**	**SAT**	**ASP**	**CBFS**	**SAT**	**ASP**
50	5,58	114,87	124,42	5,99	196,5	109,17
100	20,9	445,99	480,35	86,58	560,61	547,02
150	40,42	995,9	1100,95	243,33	1067,9	1297,69
200	52,79	1934,32	2064,22	542,03	2286,13	2257,48
250	134,05	4020,09	3486,84	757,59	3946,97	3590,85
300	190,07	6388,18	5091,78	1308,87	5067,82	5261,65
350	109,29	12889,07	7346,31	2809,9	6556,83	7293,55
400	143,5	9494,64	9187,11	2420,81	9412,54	9272,53
450	169,88	16936,5	12216,89	8242,6	12762,48	11840,33
500	247,34	19259,89	15346,48	6042,98	17155,77	14772,47
600	123,48	25139,14	22656,14	6669,8	27619,09	22382,34
700	182,4	38368,14	32421,25	7654,56	25054,57	30514,04
800	148,51	50235,33	42687,79	13854,46	36131,42	39473,17
900	318,75	97438,17	56722,6	15212,05	64424,25	52344,86
1000	298,75	87783,4	72578,13	18637,97	60564,86	65067,82

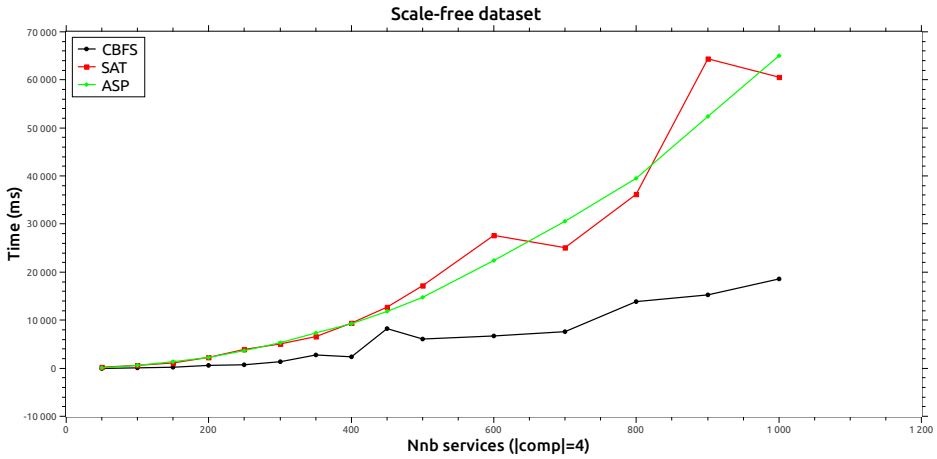
TABLE 4.7 – L'influence du nombre de services candidats par classe sur l'efficacité des approches proposées.

cul augmente au fur et à mesure que le nombre de services par classe augmente. Certaines exceptions sont faites par l'approche Max-SAT, où nous trouvons un temps de calcul plus petit avec des services candidats plus importants. Ce comportement s'explique par le fait que les approches SAT profitent de nombreuses heuristiques pour accélérer la recherche de solution. Ainsi, l'augmentation de la taille des formules booléennes générées induite par l'augmentation du nombre de services candidats n'est pas le seul facteur qui influence le temps de calcul. L'efficacité des heuristiques appliquées à une instance de problème particulier peut avoir un impact apparent sur le temps de calcul. Cette expérimentation confirme notre conclusion précédente selon laquelle l'algorithme CBFS donne de meilleures performances sur les petites compositions. Ce résultat est plus apparent dans l'ensemble de données Small-world. Cela est justifié par le fait que dans une petite composition, la profondeur de l'état final dans le graphe multi-niveaux devient également petite, et donc rapidement attiend. Il est aussi important de noter que les performances des approches déclaratives sont très proches, et cela est vrai pour les deux ensembles de données.

En résumé, et selon les expérimentations précédentes, il est difficile de favoriser une approche particulière pour être utiliser dans notre Framework de sélection. Bien que l'algorithme CBFS donne de meilleures performances dans des petites compositions, il est moins évolutif dans les compositions plus grandes. Dans ces dernières, les approches déclaratives sont plus performantes. Les résultats des Expérimentation montrent aussi que les performances des approches déclaratives



(a)



(b)

FIGURE 4.14 – L'influence du nombre de services candidats par classe sur l'efficacité des approches proposées : (a) Small-world dataset, (b) Scale-free dataset.

sont très proches. Cela reste vrai malgré le petit avantage de l'approche ASP sur l'approche Max-SAT. Nous remarquons également que le solveur ASP a montré un comportement très stable, ce qui représente une caractéristique très intéressante, surtout si le système de sélection doit garantir certain de temps de réponse. Notre objectif était de démontrer l'adéquation de ces approches sur le modèle de vie privée proposé, et aussi dans le domaine de sélection de services à base de critères de vie privée. Cette adéquation nous permet de bénéficier de tous les avantages des approches déclaratives comme la flexibilité et l'extensibilité. Ces derniers nous permettent, par exemple, d'ajouter ou de supprimer facilement des contraintes sans remodeler ou réécrire tous les algorithmes de sélection. De plus, ces approches offrent la possibilité de bénéficier du progrès continu des solveurs SAT et ASP pour améliorer les performances du Framework de sélection.

4.7 Conclusion

Dans ce chapitre, nous avons présenté un modèle pour préserver la vie privée dans les services Web. Le modèle permet de préserver la vie privée des utilisateurs ainsi que les fournisseurs de services. Le modèle proposé est implémenté sur un problème de sélection de services, dont l'objectif est de trouver une composition qui réponde au mieux à toutes les exigences de vie privée des utilisateurs et les fournisseurs de services. Plusieurs algorithmes de sélection sont conçus en se reposant sur le modèle proposé. Tous ces algorithmes retournent des compositions qui respectent toutes les contraintes de vie privée et minimisent le risque d'une menace relative. Si un service viole l'une des contraintes, ce dernier sera écarté de la solution. Comme perspectives, nous envisageons d'explorer d'autres approches génériques de résolution de problèmes telles que la satisfaction modulo théories (SMT) [Barrett09]. Une autre direction prometteuse pour les travaux futurs consiste à traiter les cas où l'ensemble des solutions est vide (c'est-à-dire il n'existe pas une composition qui préserve les contraintes de vie privée). Une solution pour faire face à cette situation consiste à modifier la façon dont nos algorithmes recherchent les solutions. Par exemple, au lieu de rechercher une composition valide qui minimise le risque de vie privée, nous pouvons opter pour une approche d'optimisation multi-objectif. Ainsi, nous minimisons à la fois le risque de vie privée engendré par les compositions ainsi que le nombre de contraintes violées. Notons que cette modification peut être effectuée sans aucune adaptation du modèle de vie privée proposé. Une autre solution consiste à étendre le modèle de vie privée pour qu'il supporte un processus de négociation. La négociation est initiée par le Framework de sélection si l'ensemble de solutions est vide. Dans un tel cas, le Framework de sélection interagit avec les fournisseurs de services afin d'assouplir (relaxer) leurs contraintes de vie privée. Après cela, la recherche est relancée en donnant la priorité aux services qui ont accepté la relaxation des contraintes. Évidemment, le processus de négociation facilite et assoupli le processus d'invocation de services, mais d'un autre côté, il introduit un surplus de consommation de ressources et de temps de calcul.

"You have to fight for your privacy or you lose it."

Eric Schmidt, Executive Chairman of Google Inc.

5

Une approche à base de Machine Learning pour la protection des micro-données

▷

Dans ce chapitre nous présentons une approche de protection des micro-donnés. Contrairement aux approches précédentes, cette approche n'introduit pas des modifications sur les données originales, mais utilise les techniques de Machine Learning pour construire des modèles à partir de ces données, ces modèles sont utilisés par la suite pour générer des nouvelles données qui se diffères totalement des données originales, ce qui introduit une forte garantie de protection. Les résultats des tests concernant l'utilité des données générées sont très prometteurs et encouragent des éventuelles améliorations sur cette approche. ◁

Plan du chapitre

5.1 Introduction

Les micro-données sont définies comme étant l'ensemble des données contenant des informations sur des répondants individuels à une enquête. Les individus peuvent être des ménages ou bien appartiennent à des organisations telles que des écoles, des hôpitaux ou des entreprises. Les réponses à une enquête nationale sur la santé de la population est un bon exemple des micro-données.

L'utilisation et le partage de ce type de données est devenu une nécessité dans plusieurs domaines telles que la santé, l'administration, l'économie ainsi que la recherche et l'enseignement universitaire. Le traitement et l'analyse de ces données peut entraîner des risques sur la vie privée et la confidentialité des individus. Le fait de supprimer les attributs identificateurs (nom, prénom, numéro de sécurité sociale, ..) ne donne pas un niveau de protection approprié, [Ciriani07] a montré qu'il est possible pour un attaquant d'identifier un individu d'une manière précise par un simple rapprochement des informations contenues dans les micro-données avec les informations disponible publiquement (une liste des électeurs par exemple), de ce fait, pour éliminer tout risque d'identification il faut appliquer d'autre traitements sur les micro-données avant de les publier.

Le problème qui se pose c'est comment modifier les micro-donnés de façon a protéger les individus et en même temps garder l'utilité de ces données.

Plusieurs approches sont apparues pour résoudre cette problématique, la majorité de ces approches se basent sur l'utilisation des techniques de suppression et de généralisation [Samarati98, Meyerson04, Samarati01, Sweeney02], d'autres ajoutent de bruits [Dwork11, Nissim17] aux données originales pour renforcer la protection.

Dans ce chapitre nous présentons une approche de protection des micro-donnés. Contrairement aux approches précédentes n'introduit pas des modifications sur les données originales, mais utilise les techniques de Machine Learning pour construire des modèles à partir de ces données, ces modèles sont utilisés par la suite pour générer des nouvelles données qui se diffères totalement des données originales, cela introduit une forte garantie de protection. Les résultats des tests concernant l'utilité des données générées sont très prometteurs et encouragent des éventuelles améliorations sur cette approche.

Le reste de ce chapitre est organisé comme suit : la section 5.2 présente Un état de l'art sur la protection des micro-données . La section 5.3 introduit les détails de l'approche proposée. Les résultats des tests seront discutés dans la section 5.4. À la fin une conclusion qui résume notre travail et présente quelques perspectives.

5.2 Protection des micro-données : Etat de l'art

5.2.1 Définition

Le terme micro-données désigne un tableau de données non agrégées. Par exemple, une table de base de données relationnelle. Dans la forme la plus basique, une base micro-données est composée d'un ensemble d'attributs de la forme : Identificateurs explicites (attributs identificateurs), attributs Quasi-identificateurs, attributs non-sensibles et attributs sensibles. Les identificateurs explicites sont un ensemble d'attributs contenant des informations qui identifient explicitement le propriétaire d'un tuple dans la base. Un exemple d'attributs identificateurs est : le numéro de sécurité sociale (SSN), le numéro d'identification national, le nom, etc. Les attributs quasi-identificateurs sont un ensemble d'attributs, qui, l'aide d'informations externes (d' autres bases) deviennent susceptibles d'identifier un propriétaire de tuple. Un exemple de tels attributs est : le sexe, la date de naissance, le code postal, etc. Les attributs sensibles comprennent des informations sensibles telles que la maladie, le revenu et le statut d'invalidité, etc. Les attributs non sensibles comportent tous les attributs qui ne font pas partie des trois catégories précédentes. La frontière entre les attributs non sensibles et quasi-identificateurs n'est pas toujours claire. Par la suite nous utilisons ces deux notions indifféremment. La

Attributs Identificateurs | Attributs non sensibles + Quasi-identificateurs | Attribut sensible

SSN	Nom	DN	Sexe	Zip	Etat civil	Maladie
123456	Albert.c	64/04/12	F	94142	divorcé	Hypertension
987654	Lee. J	64/09/13	F	94141	divorcé	obésité
098765	Chan .C	64/04/15	F	94139	marié	Douleur à la poitrine
...	…	63/03/13	H	94139	marié	obésité
...	…	63/03/13	H	94139	marié	soufflecourt
...	…	63/03/18	F	94138	célibataire	soufflecourt
...	…	64/09/27	F	94141	veuve	soufflecourt

FIGURE 5.1 – Exemple de table Micro-données.

figure 5.1 présente un exemple de table micro-données. Les données de cet exemple comportent deux attributs identificateurs (SSN et Nom), des attributs quasi identificateurs et non sensibles(la date de naissance, le sexe, le zip code et l'état civil) et un seul attribut sensible qui est la maladie.

5.2.2 Modèles d'attaques sur les Micro-données

Dans cette section, nous introduisons quelques types d'attaques sur les micro-données. Nous présenterons en particulier, les attaques de types : liaison d'enregistrements, liaison d'attribut, liaison de tables et les attaques probabilistes. Tous ces types d'attaques font partie des attaques par inférence déjà discutées au chapitre 2, section 2.6.4. Les définitions dans cette section sont basées principalement sur le travail de Wang et al. [Wang10].

5.2.2.1 Attaque par liaison d'enregistrements (Record Linkage)

Une attaque de type liaison d'enregistrements aura lieu dans le cas où une certaine valeur d'un attribut quasi-identificateur identifie un petit nombre d'enregistrements dans la table publiée. Si pour un individu la valeur de son attribut quasi-identificateur correspond à cette valeur, il sera susceptible à être liée à un petit nombre d'enregistrements. Dans ce cas, l'attaquant ne fait face qu'à un petit nombre de possibilités pour identifier l'enregistrement de la victime, et avec l'aide de connaissances supplémentaires, il y a une chance que l'attaquant puisse identifier de manière unique l'enregistrement de la victime.

5.2.2.2 Attaque par liaison d'attribut (Attribute Linkage)

Une attaque de type liaison d'attribut peut se produire dans le cas où certaines valeurs sensibles prédominent dans un groupe, une inférence réussie devient relativement facile même avec des mécanismes de protection comme le K-anonymat. Dans ce type d'attaque, l'attaquant peut ne pas identifier précisément l'enregistrement de la victime cible, mais peut inférer ses valeurs sensibles à partir des données publiées, en se basant sur l'ensemble des valeurs sensibles associées au groupe auquel la victime appartient. Ce type d'attaque est connu aussi sous le nom d'attaque d'homogénéité .

5.2.2.3 Attaque par liaison de table (Table Linkage)

Une attaque de type liaison de table se produit si un attaquant peut déduire la présence ou l'absence de l'enregistrement d'un individu dans la table publiée. Avoir cette information peut révéler dans certains cas des informations très sensibles. L'exemple le plus évident pour ce type d'attaque, est le cas d'un hôpital qui publie une table de données avec un type particulier de maladie. L'identification de la présence d'un individu particulier dans la table est déjà dommageable. Ce type d'attaque peut être considéré un type de ce qu'on appel « background attack », une attaque qui se base sur l'utilisation des informations préalables pour inférer

des nouvelles informations après l'accès aux données publiées. Une approche qui protège contre des attaques de type liaison de table, assure une protection implicite contre les attaques de type liaison d'enregistrements et d'attributs, du fait qu'un attaquant qui utilise ces deux derniers types suppose déjà l'existence de la cible dans la table publiée.

5.2.2.4 Attaque Probabiliste

Une attaque probabiliste peut se produire si un attaquant aura la capacité de changer ses croyances probabilistes sur les informations sensibles d'un individu après avoir accéder aux données publiées. En général, les approches qui lutent contre des attaques probabilistes essaient de minimiser la différence entre les croyances probabilistes antérieurs et postérieures.

5.2.3 Les techniques de protection des micro-données

Cette section présente quelques techniques d'anonymisation des micro-données. Ces techniques sont à la base des différentes approches de protection introduites dans la section suivante.

5.2.3.1 Généralisation et Suppression

L'objectif de cette technique est de cacher certains détails des attributs quasi-identificateurs. La généralisation, consiste à remplacer une valeur donnée par une valeur plus générale selon une hiérarchie donnée, si l'attribut est catégorial, et par un intervalle si l'attribut est continu. Dans le cas où la généralisation est impossible, la valeur sera supprimée.

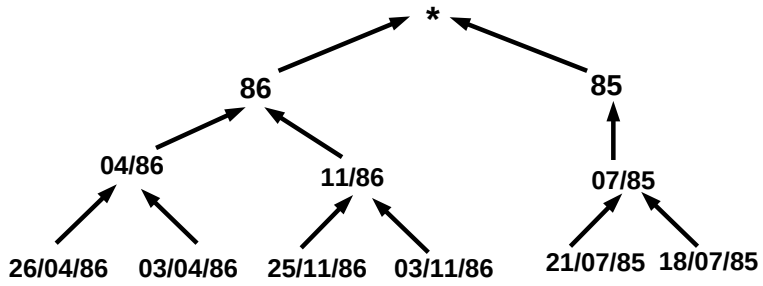


FIGURE 5.2 – Exemple d'hiérarchie de généralisation de l'attribut « date de naissance »

La figure 5.2 représente un exemple d'hiérarchie de généralisation de l'attribut « date de naissance ». Chaque niveau dans cette hiérarchie révèle moins de détails

sur la date de naissance. La racine de la hiérarchie ne révèle aucun détail sur l'attribut, ce qui correspond à une opération de suppression.

5.2.3.2 Perturbation

C'est une famille de techniques qui se basent sur la publication des valeurs perturbées (modifiées) d'une table au lieu des valeurs exactes. La perturbation est faite de sorte que les informations statistiques calculées à partir des données perturbées ne diffèrent pas significativement des informations statistiques calculées à partir des données originales [Domingo08]. Cette famille inclut plusieurs sous techniques, comme : *Le bruit additif, le Swapping, le Ré-échantillonnage et l'Arrondissement.*

1. **Le Bruit additif** :
   L'idée générale est de remplacer une valeur sensible originale s par s + r , où r est une valeur aléatoire tirée d'une certaine distribution. Notons que cette technique ne convient pas pour protéger les données catégorielles, mais bien adaptée aux données continues. Pour cette raison, qu'elle est utilisée souvent pour cacher des données numériques sensibles (par exemple, le salaire) [Duncan00].
2. **Le Swapping** :
   Le principe de cette technique est d'anonymiser une table en échangeant (le swap) les valeurs des attributs sensibles entre les enregistrements. Les swaps sont faits d'une manière à maintenir certaines propriétés pour des analyses statistiques. Contrairement à la technique du bruit additif, cette technique peut être utilisée pour les données numériques ainsi que catégorielles [Reiss84].
3. **Le Ré-échantillonnage** :
   Technique proposée initialement par [Heer93], cette technique se base sur la création d'un nombre « m » d'échantillons $E_1, E_2, \ldots, E_m$ indépendants des valeurs d'un attribut original X_i. Les échantillons crées seront triés avec le même critère de classement. En suit, l'attribut perturbé $X_i^{'}$ est construit à partir des valeurs $\overline{x}_1, \ldots, \overline{x}_n$, où n est le nombre d'enregistrements et $\overline{x}_j$ est la moyenne des valeurs de rang j dans $E_1, E_2, \ldots, E_m$. Cette technique est applicable que sur les données numériques.
4. **L'Arrondissement** :
   Cette technique se base sur le remplacent des valeurs originales des attributs par des valeurs arrondies. Pour un attribut donné Ti , les valeurs arrondies sont choisies parmi un ensemble de points d'arrondissement défini à l'avance. Cet ensemble est souvent défini en utilisant les multiples d'une

valeur de base « b ». L'étape d'arrondissement consiste à définir une fonction qui translate chaque valeur original de l'attribut Ti, vers une valeur de l'ensemble d'arrondissement [Domingo01]. Cette technique est applicable seulement sur les données numériques.

5.2.4 Les Approches de Protection

Dans cette section nous présentons quelques approches de protection des micro-données. Nous avons concentré sur les approches les plus citées dans la littérature comme le modèle K-anonymat, un état de l'art plus complet sur les approches de protection est présenté dans [Wang10] et [Charu08].

5.2.4.1 Le modèle K-anonymat

Le modèle K-anonymat [Samarati98] est l'une des solutions les plus populaires dans ce domaine, elle se repose sur les opérations de généralisation et de suppression. Son objectif est de ne publier des données que s'il y a au moins k individus identiques dans chaque groupe de données généralisées.

L'obtention d'une solution k-anonymat optimale a été prouvée comme étant un problème NP-difficile [Meyerson04], ainsi plusieurs algorithmes et méthodes ont été proposés. Certains de ces algorithmes sont basés sur des méthodes complètes qui utilisent des approches de recherche gloutonnes, d'autres sont basées sur des méthodes heuristiques.

Samarati [Samarati01] a proposé un algorithme complet pour identifier tous les k-minimal généralisations et choisit le k-anonymat optimale parmi eux en fonction de certains critères. Pour réduire l'espace de recherche, il a utilisé certaines propriétés des treillis de généralisation comme la monotonie.

Sweeney [Sweeney02] utilise un algorithme exhaustif qui explore toutes les possibilités de généralisation pour choisir la solution optimale. Il est clair que les deux algorithmes précédents ne sont pas pratiques si la taille des données publiées est grande. Bayardo et Agrawel [Bayardo05] ont proposé un algorithme optimal. Au lieu d'utiliser l'ensemble de données originales et rechercher la généralisation optimale, ils ont commencé à partir de l'ensemble de données les plus générales et ils ont cherché une spécialisation optimale en utilisant des indicateurs de coûts bien définis.

Aussi plusieurs algorithmes heuristiques ont été proposés. Iyengar [Iyengar02] a utilisé un algorithme génétique pour explorer le vaste espace de recherche de généralisations. Il a proposé également une définition très souple de la notion de généralisation d'attributs. Lunacek et al. [Lunacek06] ont amélioré l'approche

d'Iyengar par l'introduction d'un nouvel opérateur de croisement pour contraindre les généralisations et éliminer les inutiles. Dewri et al. [Dewri08] ont formulé le k-anonymisation comme un problème d'optimisation multi-objectif pour accomplir le compromis entre le niveau de protection et la perte des données. Ils ont utilisé l'algorithme NSGA-II pour résoudre le problème d'optimisation multi-objectif relevé. Run et al. [Run12] ont utilisé une approche hybride qui combine entre la recherche Tabu et les algorithmes génétiques pour améliorer la performance du processus de recherche.

Nadimpalli et al. [Nadimpalli11] ont proposé une approche pour réduire les associations entre les quasi-identificateurs et attributs sensibles par décomposition d'un tableau de k-anonyme dans un tableau quasi-identificateurs et un ou plusieurs tableaux sensibles. Cette solution n'est utile que si l'ensemble de données diffusées contient plus d'un attribut sensible.

Morton et al. [Morton12] ont proposé un algorithme de Clustering conduit par une nouvelle mesure d'utilité basée sur l'ajout de la notion de règles de contrainte de données qui, si elle est maintenue, aide à maximiser l'utilité des données anonymes.

5.2.4.2 (p-sensitive, k-anonymity)

Une approche proposée par [Truta06], elle est considérée comme une évolution de l'approche K-anonymity. La protection dans cette approche vise à éviter la divulgation d'attributs en exigeant au moins « p » valeurs différentes pour chaque attribut sensible au sein des tuples partageant une combinaison de quasi-identifieurs. On dit qu'une tables T satisfait (k -anonymie p-sensible) , pour $k > 1$ et $p \leqslant k$, si pour chaque classe d'équivalence, le nombre de valeurs distinctes pour chaque attribut sensible est au moins p. Cependant, l'approche (p-sensitive, k-anonymity) a la limitation de supposer implicitement que chaque attribut sensible prend des valeurs uniformément réparties sur son domaine, c'est-à-dire que les fréquences des différentes valeurs d'un attribut sensible sont similaires, ce qui n'est pas toujours le cas.

5.2.4.3 L-diversité

Machanavajjhala et al. [Machanavajjhala07] ont montré que le K-anonymat est vulnérable à des attaques appelées d'homogénéité et de connaissances antécédentes. Ils ont proposé le concept de L-diversité qui rende une table de données anonyme d'une façon que dans chaque groupe de quasi-identificateurs il y a au plus " 1/L " enregistrements qui possèdent la valeur sensible la plus fréquente. Le (α , k)-anonymat [Wong06] est la combinaison du k-anonymat et le l-diversité. Cette

solution protège à la fois les données sensibles et d'identification en réduisant l'attaque d'homogénéité. Les auteurs dans [Li07] ont montré que le modèle L-diversité gère mal le cas où la distribution globale d'un attribut sensible est biaisée (skewed), ce qui peut exposer les individus à des menaces sur leurs vie privée.

5.2.4.4 t-closeness

Le modèle de t-closeness a été proposé par Li Ninghui et al.[Li07] . Ce modèle utilise la propriété selon laquelle la distance entre la distribution de l'attribut sensible dans un groupe anonymisé ne devrait pas être différente de la distribution dans la table entière de plus d'un seuil "t". Les auteurs dans ce travail ont utilisé la métrique Earth Mover Distance[Rubner00] pour quantifier la distance entre les deux distributions. Malgré que ce modèle résout quelques problèmes du modèle l-diversity comme la vulnérabilité de divulgation d'attribut, il provoque une forte dégradation sur l'utilité des données publiées.

5.2.4.5 Differential Privacy

Le concept " differential Privacy " [Dwork11, Nissim17] a été introduit récemment comme modèle de protection de données personnelles. Ce modèle est largement adopté par la communauté scientifique du fait qu'il assure une protection stable et indépendante des connaissances a priori et a posteriori (avant et après l'accès à des données publiées) de l'attaquant sur un individu particulier. Autrement dit La " differential Privacy " exige que le risque qu'un individu soit victime d'une attaque sur sa vie privée ne devrait pas augmenter sensiblement à la suite de la participation à une base de données statistiques. La majorité des méthodes proposées pour réaliser une "differential Privacy" utilisent une table de contingence en ajoutant un bruit sur la fréquence de chaque groupe. La table de contingence est issue aussi d'une généralisation de la table des données originales.

5.2.4.6 Génération de données synthétiques

Cette approche est la plus proche de l'approche proposée dans ce chapitre. Le principe est de construire un modèle statistique à partir des données originales, puis d'échantillonner des points du modèle. Ces points échantillonnés forment les données synthétiques qui seront publiées au lieu des données originales [Rubin93]. Une autre approche de génération de données synthétiques a été proposée par [Aggarwal08]. L'idée est de condenser d'abord les enregistrements en plusieurs groupes. Pour chaque groupe, extraire des informations statistiques, telles que la somme et la covariance, pour préserver la moyenne et les corrélations entre les dif-

férents attributs. Ensuite, générer des points pour chaque groupe en fonction des caractéristiques statistiques du groupe. D'autres algorithmes de génération de données synthétiques ont été créés en faisant varier le modèle synthétique construit à l'aide des données originales. Par exemple, les auteurs dans [Fienberg98] ont proposé une méthode à base d'une technique de Bootstrap [Efron92]. Cette approche se base sur l'introduction de certaines modifications sur une fonction de distribution construite à partir des données originales. En suite cette fonction est utilisée pour générer les données synthétiques.
Sachant que la génération des valeurs synthétiques pour tous les attributs d'une base de données peut être difficile dans la pratique. Ainsi, plusieurs auteurs [Burridge03, Reiter03, Reiter05, Abowd01] ont opté pour un mixage entre des données réelles et données synthétiques. Cette approche a l'avantage de produire des données qui ont une utilité meilleurs, mais par contre une plus faible protection que les données entièrement synthétiques.

Au meilleur de nos connaissances, il n'existe aucune approche qui utilise des techniques de machine Learning pour la génération des données synthétiques, et des règles sémantiques pour capturer les corrélations entre ces données.

Le table 5.1 donne un résumé sur les approches présentées et les attaques visées par ces approches.

Approche	**Modèle d'attaque**			
	Liaison d'enregistrements	Liaison d'attribut	Liaison de table	Attaque probabiliste
K-anonymat	✓			
(p-sensitive, k-anonymity)	✓	✓		
L-diversité	✓	✓		
(α, k)-anonymat	✓	✓		
t-closeness	✓	✓		✓
Differential Privacy			✓	✓
Données synthétiques	✓	✓	✓	✓

TABLE 5.1 – Les techniques d'attaques visées par les approches de protection.

5.3 L'approche proposée

L'idée principale de l'approche proposée est de publier des données fictives au lieu de vrais données qui exposent les individus à des risques d'attaques sur leur confidentialité. Les données fictives sont générées en utilisant des modèles issues des données originales, ce qui permet aux nouvelles données de garder certaines propriétés des données originales. L'approche se compose de deux étapes principales : *la génération des données*, et *l'évaluation des données générées* (figure 5.3).

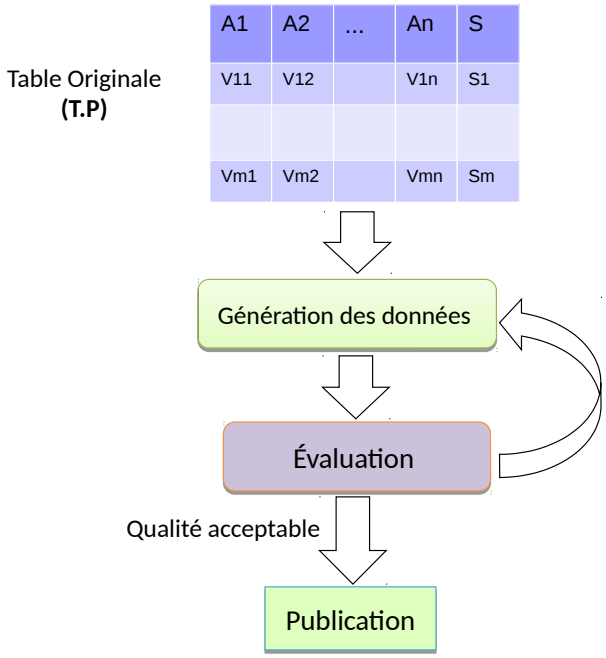


FIGURE 5.3 – Le principe de l'approche

L'étape d'évaluation permet de tester la qualité des données générées. Si la qualité n'est pas satisfaisante on refait l'étape de génération. La suite de cette section introduit une brève formulation du problème, avant d'expliquer les étapes de génération des données ainsi que le mécanisme d'évaluation.

5.3.1 La formulation du problème

Soit une table *" TP "* dite table privée qui illustre les données originales d'un ensemble de personnes. La table *" TP "* contient *" N "* attributs, ces attributs sont divisés en deux catégories : les attributs non sensibles :$A = A1, A2, ...Am$ (ex : âge, sexe, niveau d'études,..) et les attributs sensibles : $S = S1, S2, ...Sk$ (ex : maladie, salaire,...), avec $m + k = N$. Dans notre modèle d'attaque, un attaquant essaie d'acquérir les valeurs des attributs sensibles d'un individu en se basant sur ses connaissances partielles sur les attributs non sensibles.

L'objectif de notre approche est de générer à partir de la table *" TP "* une nouvelle table *" TG "* dite générée, cette dernière sera publiée à la place de la première. La table *" TG "* doit contenir exactement le même nombre d'attributs. Dans notre travail nous avons traité le cas d'un seul attribut sensible, la généralisation sur plusieurs attributs sensibles n'affecte pas les principes de notre approche.

5.3.2 La génération des données

La génération de la table " TG " passe par trois étapes (figure 5.4) :La construction d'un modèle de classification, la génération des attributs non sensibles, et La prédiction de l'attribut sensible.

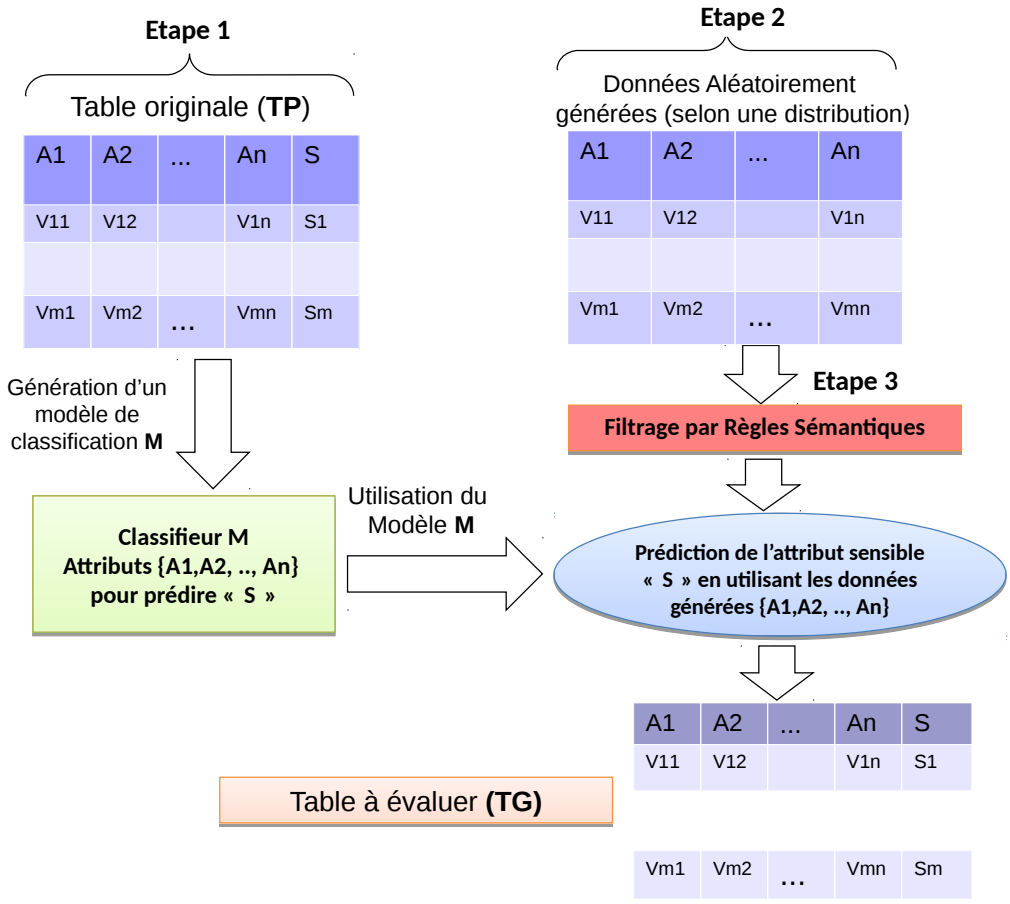


FIGURE 5.4 – Les étapes de génération des données

Etape 1 : La construction du modèle de classification

Dans cette étape un modèle de classification est construit à partir de la table " TP ". Ce modèle est issu d'un mécanisme d'apprentissage sur les données de la table " TP ". A la fin de cette étape on obtient un modèle capable de prédire la valeur de l'attribut sensible " S " en prenant comme entrés les attributs non sensibles (A1...An).

Etape 2 : La génération des attributs non sensibles

Cette étape consiste à générer aléatoirement un ensemble de valeurs pour chaque attribut non sensible en utilisant les intervalles des valeurs issues des données originales. Cette étape est guidée par un ensemble de règles sémantiques appliquées sur les données générées pour éviter les cas réelle-

ment impossibles (voir figure 5.5. Un exemple de règle sémantique est : " un enfant ne peut être marié", " un enfant ne peut avoir d'enfants ",...

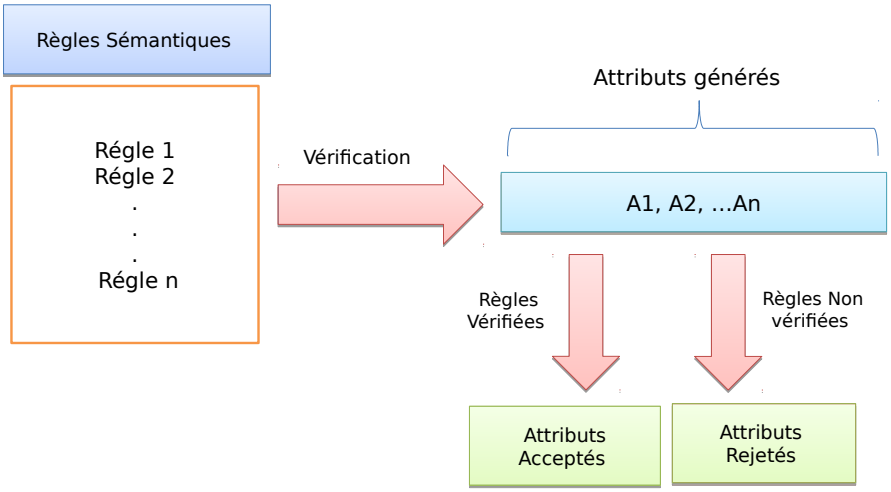


FIGURE 5.5 – Filtrage par Régles Sémantiques.

Etape 3 : La prédiction de la classe sensible

En utilisant le modèle de classification construit à l'étape 1 et les données générées à l'étape 2, on peut prédire les différentes valeurs de l'attribut sensible " S ". La fin de cette étape produit la table " TG ". Si cette table est approuvée par le mécanisme d'évaluation, elle sera publiée au lieu de la table " TP ".

5.3.3 Le mécanisme d'évaluation

Dans notre approche nous avons testé la qualité des données générées selon deux aspects. Dans le premier aspect, nous avons testé la qualité des classifieurs issus en utilisant les données générées, ce test est utile dans le cas ou les données générées seront utilisées pour un but de classification. Dans le deuxième aspect, nous avons testé la capacité des données générées à produire des règles d'association. ce test évalue la qualité des données dans le cas ou ces données seront utilisées dans un processus de data-mining. Les deux sous-sections suivantes introduisent les détails de chaque mécanisme d'évaluation.

5.3.3.1 Évaluation pour un but de classification

Le mécanisme d'évaluation consiste à comparer les performations d'un modèle de classification issu des données originales (table TP), avec un modèle issu des données générées (table TG), si la différence ne dépasse pas un certain seuil on peut valider les données générées. Les étapes de l'évaluation se déroulent comme suit (figure 5.6) :

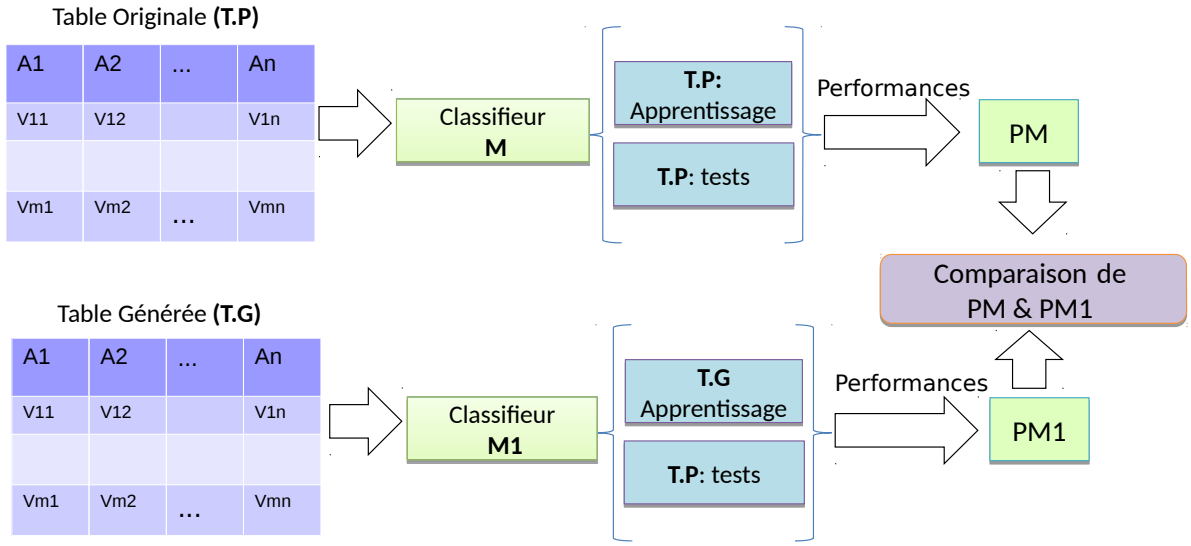


FIGURE 5.6 – Le mécanisme d'évaluation pour un but de classification.

Etape 1 : Dans cette étape on construit un modèle de classification "M" à partir de la table "TP". Cette construction se base sur l'utilisation d'une partie des données de la table "TP" comme base d'apprentissage. Les performances du modèle construit notées "PM" sont tirées en utilisant une partie de la même table notée "Ttest" comme base de tests.

Etape 2 : Dans cette étape on construit aussi un modèle de classification "M1" à partir de la table générée "TG". Cette construction se base sur l'utilisation des données de la table " TG " comme base d'apprentissage. Les performances du modèle construit, notées " PM1 " sont tirées en utilisant le même jeux de test "Ttest" que l'étape précédente.

Etape 3 : cette étape compare " PM " avec " PM1 ". Si les différences ne dépassent pas un certain seuil (ex : la différence de la précision de " M " et " M1 " ne dépasse pas un seul S), les données générées peuvent être validées.

5.3.3.2 Évaluation pour un but de data-mining

La validation des données générées pour un but de data-mining est faite en mesurant le nombre de règles d'association communes extraites à partir des données originales et les données générées (voir figure 5.7). Plus ce nombre est grand plus la qualité des données générées et bonne. Le choix de l'utilisation des règles d'association ((voir l'annexe , section)) comme critère de validation est justifié par l'importance de ces règles dans la découverte des relations significatives entre attributs. En plus le processus d'extraction des règles d'association englobe un autre aspect important de data-minig qui est l'extraction des motifs fréquents.

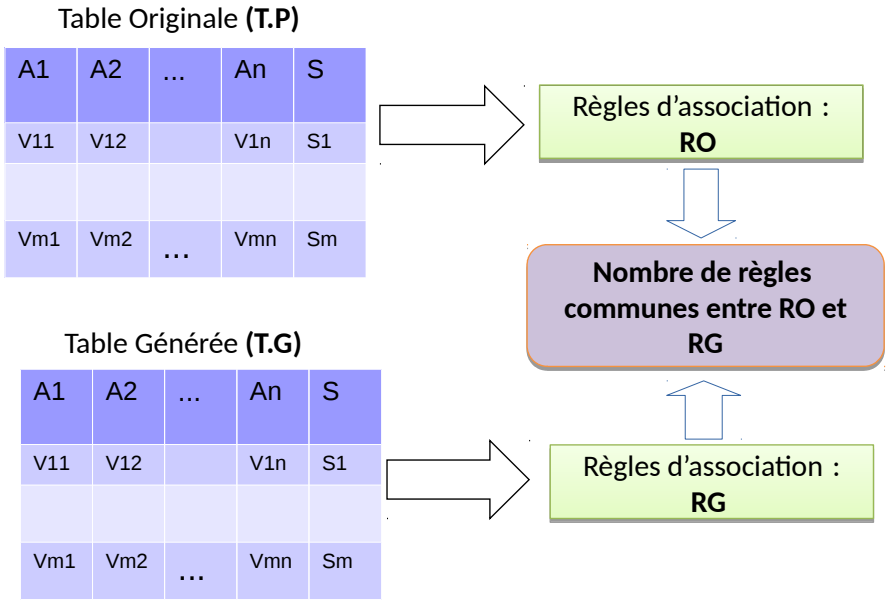


FIGURE 5.7 – Le mécanisme d'évaluation pour un but de Data-mining.

5.4 Expérimentation

Cette section présente quelques expérimentations pour évaluer l'approche proposée. En plus de l'évaluation de la qualité des données générées, nous avons étudié l'impact de l'utilisation des règles sémantiques sur la qualité des données générées.

5.4.1 Description de la Base

Nous avons testé notre approche sur la base "Adult Data Set" [1]. La base contient 14 attributs dont 5 sont numériques, et 8 sont nominales dont une variable binaire indexant le revenu annuel d'un individu (<=50K ou > 50K). Ce dernier représente l'attribut classe pour la base et l'attribut sensible dans nos tests. Les attributs numériques sont : l' âge (age), les heures travaillées par semaine (hours-per-week), le code du niveau d'étude (education-num), le gain en capital (capital-gain), la perte en capital (capital-loss) et un attribut de poids l'enquête (fnlwgt) qui représente un score démographique attribué à un individu. Les attributs nominales sont : le sexe (sex), la race (race), le pays d'origine (native-country), le niveau d'étude (education), la relation familiale (relationship), la classe d'emploi (workclass), la profession (occupation) et l'état civil(marital-status).

1. http://archive.ics.uci.edu/ml/datasets/Adult

Les données de la base (environ 30000 enregistrement après suppression des données manquantes) ont été divisées aléatoirement en deux ensembles : un ensemble d'apprentissage portant sur 66% de la totalité de la base, et un ensemble de test portant sur les 33% des enregistrements restants.

5.4.2 La génération des données

Du fait que l'attribut sensible dans notre base est un attribut binaire (le revenu >50K ou <=50K), nous avons choisi pour la construction du modèle de classification dans la phase de génération des données (voir section 5.3.2) un classifieur de type "Support Vector Machines" (SVM) [Chang11], un classifieur très adapté pour une classification binaire.

Notons que les attributs non-sensibles et quasi-identificateurs de type numérique (age, heurs de travail, le poids de l'enquête,gain et perte en capitale, et numéro d'études) sont générés en respectant la moyenne et l'écart-type des données originales.

Pour le filtrage à base de règles sémantiques, nous avons utilisé un total de 30 règles. La figure 5.8 donne un exemple des règles sémantiques utilisées.

R1: (relationship = "Husband") ∧ ¬(sex = "Male") ⇒ ⊥

R2: (relationship = "Wife") ∧ ¬(sex = "Female") ⇒ ⊥

R3: (marital_status = "Never-married") ∧ ((relationship = "Wife") ∨ (relationship="Husband")) ⇒ ⊥

R4: (marital_status = "Married-civ-spouse") ∧ ¬((relationship ="Wife") ∨ (relationship="Husband")) ⇒ ⊥

R5 : (education = "Preschool") ∧ ((workclass = "Federal-gov") ∨ (workclass = "Local-gov") ∨ (workclass = "State-gov")) ⇒ ⊥

R6 : (education = "Preschool") ∧ ¬(education_num = "1")) ⇒ ⊥

R7: (education = "1st-4th") ∧ ¬(education_num = "2") ⇒ ⊥

FIGURE 5.8 – Exemple de règles sémantiques utilisées dans le filtrage.

Dans cet exemple, si pour un tuple, les prémisses de l'une des règles sont vérifiées, alors ce tuple sera supprimé. Par exemple dans la règle R1, si l'attribut « relationship » prend la valeur « Husband », et l'attribut « sex » prend une valeur différente de « male », alors le tuple est invalide et il sera supprimé de la base.

Pour garder le même nombre d'enregistrements ainsi que le même pourcentage de l'attribut classe que ceux des données originales (20000 enregistrements, 30% >50k, 70% <=50k), nous avons générées aléatoirement 100000 enregistrements. Pour avoir 20000 enregistrements de la base sans règles sémantiques, nous

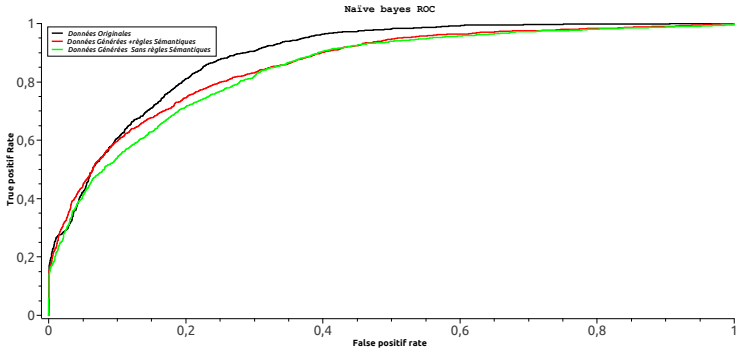
avons appliqué sur les 100000 enregistrements un échantillonnage aléatoire. Pour avoir la base générées avec règles sémantiques, nous avons appliqué sur les même 100000 enregistrements un filtrage à base de règles sémantiques, puis un échantillonnage pour ne garder que 20000 enregistrements.

5.4.3 Évaluation pour un but de classification

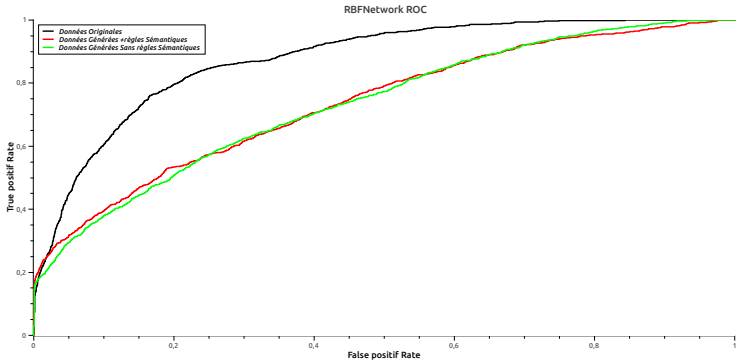
Comme nous l'avons déjà mentionné, cette expérimentation compare les performances d'un modèle de classification issue des données originales, avec un modèles issues des données générées selon l'approche proposée. Pour les données générées nous avons évalué aussi l'influence de l'utilisation des règles sémantiques sur la qualité des données générées. À cet effet, nous avons utilisé quatre algorithmes : Naive bayes [Zhang05], RBF Network (radial basis function networke) [Broomhead88], J48 [Quinlan14] et les tables de décision [Kohavi95]. L'utilisation de plusieurs algorithmes renforce le mécanisme de test et donne plus de crédibilité aux données générées. Notre choix des algorithmes est fait d'une manière à valider les données générées sur plusieurs familles d'algorithme : la famille des classifieurs bayésiens est représentée par l'algorithme Naive bayes, la famille des classifieurs fonctionnels est représentée par l'algorithme RBF Network, la famille des classifieurs arbres de décisions est représentée par l'algorithme RJ48, et en fin la famille des algorithmes à base de règles est représentée par les tables de décision .

Pour évaluer les performances des modèles issues des données générées nous avons utilisé les courbes ROC (Receiver Operating Characteristic) [Fawcett06]. Dans une telle courbe, le taux de vrais positifs (sensibilité) est tracé en fonction du taux de faux positifs (spécificité) pour différents points de coupure. Chaque point de la courbe ROC représente une paire sensibilité/spécificité correspondant à un seuil de décision particulier. Un test avec une discrimination parfaite (pas de chevauchement dans les classes) a une courbe ROC qui traverse le coin supérieur gauche (sensibilité de 100%, spécificité de 100%). Par conséquent, plus la courbe ROC est proche du coin supérieur gauche, plus les performances du classifieur sous-jacent sont meilleurs [McNeil84]. Une courbe ROC peut être réduite à une valeur unique, c'est l'AUC (Area Under ROC Curve). Cette valeur est obtenue en calculant la surface sous la courbe ROC, et elle permet de mesurer les performances de plusieurs classifieurs pour évaluer quel modèle est meilleur en moyenne [Bradley97].

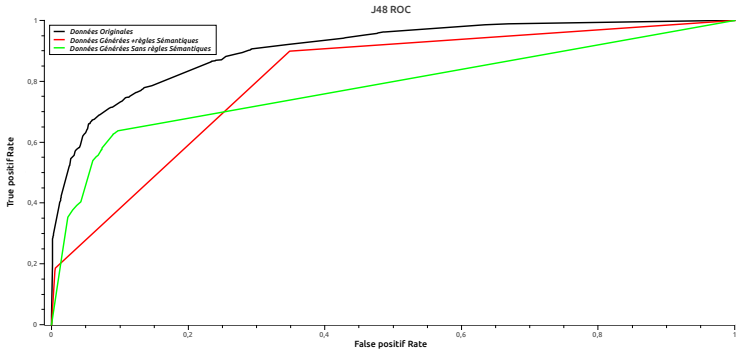
La figure 5.9 représente les courbes ROC relatives aux modèles issues des données originales, les données générées sans règles sémantiques et les données générées avec règles sémantiques et cela pour chaque classifieur (5.9(a) Naïve bayes, 5.9(b) RBFNetwork, 5.9(c) J48 et 5.9(d) Tables de Décision) .



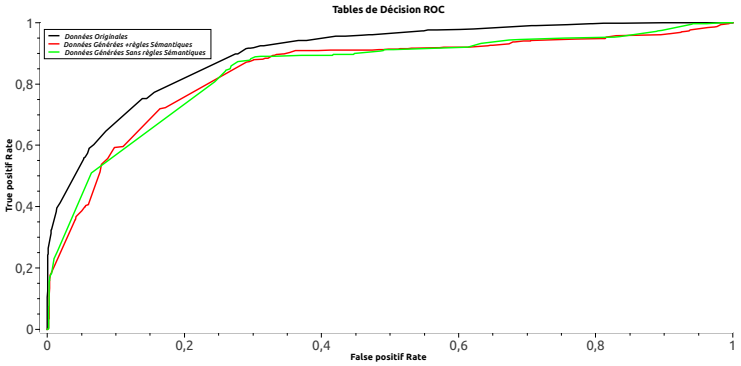
(a)



(b)



(c)



(d)

FIGURE 5.9 – Les courbes ROC relatives aux données originales, données générées avec et sans règles sémantiques pour chaque algorithme.

Les courbes ROC montre clairement la supériorité des modèles issus des données originales par rapport aux modèles issues des données générées (avec ou sans règles sémantiques). On note aussi une légère dominance des modèles construits avec les données générées avec les règles sémantiques sur les modèles construits sans règles sémantiques. Ces remarques sont confirmées avec les valeurs AUC de chaque classifier. Ces valeurs sont représentées dans la figure 5.10 ainsi que les tables : 5.2 pour Naïve bayes, 5.3 pour RBFNetwork, 5.4 pour le J48 et 5.5 pour les tables de décision.

Naïve bayes	Taux de TP	Taux de PF	AUC
Données Originales	0.832	0.38	**0.889**
Données Générées sans règles sémantiques	0.758	0.759	**0.815**
Données Générées avec règles sémantiques	0.803	0.555	**0.821**

TABLE 5.2 – Comparaison des performances des modèles issus des données originales et les données générées pour l'algorithme Naïve bayes.

RBFNetwork	Taux de TP	Taux de PF	AUC
Données Originales	0.834	0.369	**0.875**
Données Générées sans règles sémantiques	0.787	0.622	**0.799**
Données Générées avec règles sémantiques	0.807	0.48	**0.814**

TABLE 5.3 – Comparaison des performances des modèles issus des données originales et les données générées pour l'algorithme RBFNetwork.

J48	Taux de TP	Taux de PF	AUC
Données Originales	0.855	0.298	**0.883**
Données Générées sans règles sémantiques	0.814	0.437	**0.764**
Données Générées avec règles sémantiques	0.818	0.476	**0.805**

TABLE 5.4 – Comparaison des performances des modèles issus des données originales et les données générées pour l'algorithme J48.

Tables de Décision	Taux de TP	Taux de PF	AUC
Données Originales	0.847	0.349	**0.894**
Données Générées sans règles sémantiques	0.815	0.487	**0.809**
Données Générées avec règles sémantiques	0.81	0.494	**0.817**

TABLE 5.5 – Comparaison des performances des modèles issus des données originales et les données générées pour l'algorithme Tables de Décision.

Selon ces résultats, nous notons que les performances des modèles issus des données générées avec des règles sémantiques sont meilleures que celles géné-

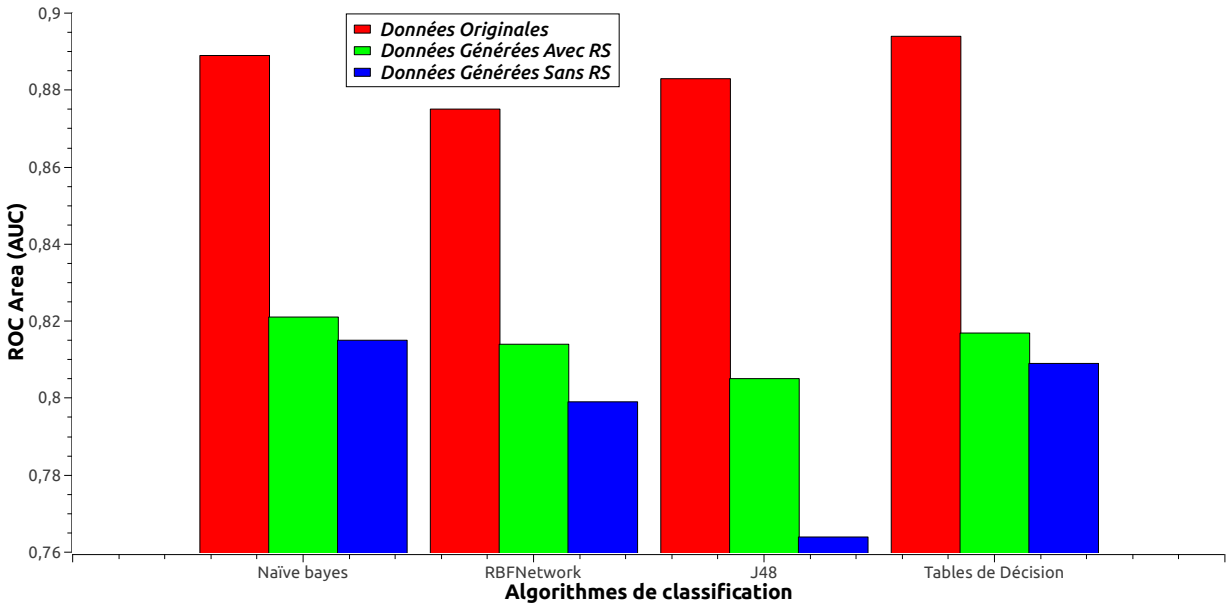


FIGURE 5.10 – Comparaison des performances des modèles issus des données originales et les données générées.

rées sans règles sémantiques, et cela pour tous les classifieurs. En effet, Pour les classifieurs Naïve bayes et les Tables de Décision les performances sont améliorées de mois de 1% (0.6 % pour Naïve bayes et 0.8% pour les Tables de Décision). L'amélioration est de l'ordre de 1.5% pour le classifieur RBFNetwor, tandis qu'elle atteint les 4% pour le classifieur J48. Malgrès que les améliorations sont légères et non signifiantes pour certains algorithmes, il est claire que la qualité des données générées avec les règles sémantique est meilleure. En effet, le filtrage par règles sémantiques élimine toute contradiction et incohérence qui existe dans les enregistrements après la génération aléatoire. Cela rend les données plus proches de la réalité, et donne plus de motivations et de confiance pour l'utilisation de ces données. Par exemple, un chercheur qui trouve dans une base q'un mari est de sexe féminin, ou bien un mari qui n'a jamais marié, a tendance à ne pas faire confiance des résultats issus de l'utilisation de cette base.

Si nous comparons ces performances avec les modèles issus des données originales, nous remarquons une claire supériorité des modèles issues de ces dernières. Les résultats montrent des dégradations des performances des modèles issus des données générées avec règles sémantiques qui se rapprochent de 7% pour les classifieurs Naïve bayes et RBFNetwork (6.8% pour Naïve bayes, et 6.1% pour RBFNetwork), et de l'ordre de 8% pour les classifieurs J48 et tables de décision (7.8% pour J48, et 7.7% pour les tables de décision). Nous jugeons que le niveau de dégradation est acceptable et qu'il reflète bien le compromis à faire entre l'utilité des données publiées et la protection de la vie privée des individus.

5.4.4 Évaluation pour un but de Data-mining

Dans cette expérimentation, nous étudions la qualité des données générées pour un but de data-mining. À cet effet, nous avons comparé le nombre de règles d'associations communes entre celles extraites avec des données originales et celles extraites avec des données générées. Le choix des règles d'association comme objet de comparaison est déjà discuté dans la section 5.3.3.2. Dans cette expérimentation nous avons examiné aussi l'impact de l'utilisation des règles sémantiques sur la qualité des données générées.
Pour l'extraction des règles d'association, nous avons utilisé l'algorithme TopKRules [Fournier12], implémenté dans l'outil SPMF [Fournier16]. l'avantage de l'utilisation de cet algorithme, c'est de rayer le besoin de l'utilisation du support minimum, un paramètre entre 0 et 1, souvent difficile à définir. Ainsi, TopKRules permet de définir directement K, le nombre de meilleurs règles à découvrir. Pour assuré une forte précision des règles extraites, nous avons utilisé une valeur de confiance égale à 1 (voir l'annexe , section).
La table 5.6 et la figure 5.11 représentent le nombre de top K règles d'association communes extraites à partir des données originales et les données générées (avec et sans règles sémantiques) avec une valeur de K qui varie entre 10 et 150.

Top K règles d'association	**10**	**20**	**50**	**100**	**150**
Données Générées sans règles sémantiques	0	0	1	13	14
Données Générées avec règles sémantiques	1	6	19	38	39

TABLE 5.6 – Le nombre de Top K règles d'association communes entre les données générées et les données originales.

Les résultats montre un grand impact de l'utilisation des règles sémantiques sur la qualité des données générées. En effet, l'utilisation de ces règles a permis de réaliser une augmentation d'au moins de 60% dans le nombre de règles communes avec les données originales (64% d'augmentation pour le top k de 100 règles, et 65% pour le top k de 150 règles). Ce taux d'amélioration est non négligeable et reflète bien les biens fait de l'utilisation des règles sémantiques sur la qualité des données générées.

Si nous comparons la qualité des données générées avec les données originales, nous remarquons une grande dégradation dans la qualité des première. Cette dégradation varie entre 62% et 74% (70% pour 20 règles, 62% pour 50 règles, 62% pour 100 règles et 74% pour 150 règles). Nous remarquons aussi que la valeur de k n'a pas une grande influence sur le taux de dégradation.

Notons que nous avons utilisé dans les deux expérimentations les mêmes bases de données générées que celles utilisées pour les tests pour un but de classifica-

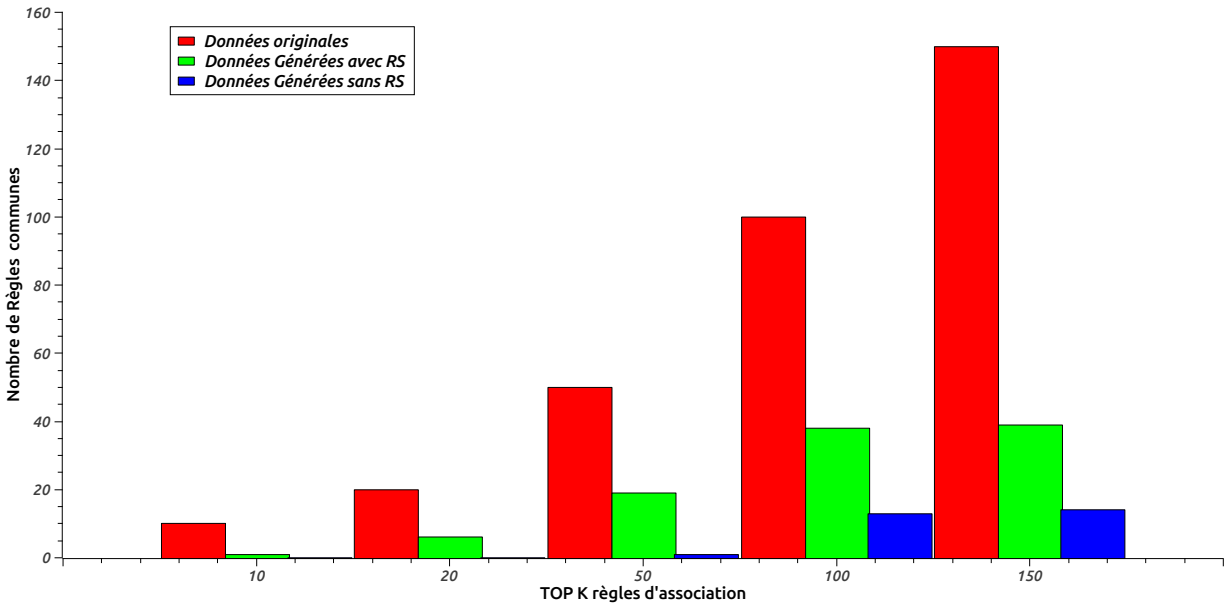


FIGURE 5.11 – Le nombre de Top K règles d'association communes entre les données générées et les données originales.

tion. Ces bases ont donné des résultats acceptables pour la classification, et des résultats moins bonnes pour la génération des règles d'association. L'une des solutions pour améliorer la qualité, est de générer des bases différentes pour chaque but, avec répétition de la phase de génération jusqu'à l'obtention des données avec une qualité acceptable. Malheureusement ce type de solution est très gourmand en temps et en ressources, et n'est pas adapté pour des systèmes qui doivent assurer certaines qualité de service comme le temps de réponse.

5.5 Conclusion

Dans ce chapitre, nous avons proposé une approche de protection de la vie privée des individus dans les bases micro-données. Cette approche génère aléatoirement des nouvelles données à partir des données originales en utilisant des classifieurs automatiques. l'utilisation de ces derniers, a permis de garder un maximum de corrélation entre les attributs sensibles et les attributs non sensibles de la base. La génération des données est renforcée par un ensemble de règles sémantiques qui rendent la base générée plus proche de la réalité et améliorent la qualité de ces données. Les nouvelles données générées se diffèrent totalement des données originales ce qui implique une grande protection de la vie privée des propriétaires des données originales.

Il est évident que l'approche proposée nécessite plus d'améliorations et d'études pour améliorer la qualité ainsi que la protection des données. Par exemple, cette approche ne traite pas le cas où certaines lignes de données générées prennent par

hasard les mêmes valeurs que celles d'individus réels. Dans ce cas, la vie privée de ces individus devient menacée. L'une des solutions à ce problème est la vérification que chaque individu dans les données générées est différent de tous les individus des données originales. Malheureusement cette vérification rend le processus de génération très lourd et surtout dans le cas des tables de données volumineuses. Aussi, l'approche proposée traite seulement le cas d'un seul attributs sensible. Ainsi, étendre notre approche pour prendre en charge plusieurs attributs sensibles est l'une de nos perspectives à court terme. Nous envisageons aussi de tester l'approche proposée pour d'autres buts comme la classification non supervisée.

"Le meilleur moyen de ne pas risquer l'échec est peut-être de commencer des projets et de ne pas les finir..."

Lyse Desroches –Romancière québécoise

6

Conclusion Générale

▷ *Conclusion, Synthèse et perspectives* ◁

Plan du chapitre

La protection de la vie privée est devenue l'une des préoccupations majeurs des utilisateurs de l'Internet. Au cours de cette thèse, nous nous sommes intéressés à traiter ce sujet sur plusieurs niveaux, touchant ainsi les différentes phases de cycle de vie des données circulants sur Internet. Notons que cette problématique est vaste et très complexe, et loin d'être traitée d'une façon complète dans une thèse.

6.1 Synthèse

Dans un premier temps, nous avons commencé par un état de l'art sur la vie privée sur Internet. l'objectif était de donner une vue générale sur ce domaine, et définir les concepts de base, avant de pouvoir entamer les trois grandes problématiques introduites dans cette thèse.

Dans la problématique de lute contre le Phishing, nous avons proposé une approche à base de liste blanche personnalisée. Le grand avantage de cette liste c'est qu'elle élimine la difficulté de la gestion et de la mise à jour de grandes quantités de données, contrairement aux approches à base de listes blanches classiques. Les faux positifs traditionnellement présents dans ces dernières sont aussi éliminés avec l'utilisation d'un classifieur automatisé comme niveau supplémentaire de filtrage. Les tests effectués sur le système proposé ont montré une amélioration par rapport à d'autres solutions.

Dans la problématique de protection de la vie privée dans les services Web, nous nous sommes concentrés sur le problème de sélection. Ce problème consiste à trouver une composition de services qui respectent les exigences de vie privée à la fois des utilisateurs et les fournisseurs de services. Un framework de sélection à été proposé dans ce cadre. Le fonctionnement de ce Framework se base sur un modèle de vie privée qui permet aux utilisateurs et aux fournisseurs de services d'exprimer sous forme de règles, leurs exigences en termes de vie privée. L'avantage de ce modèle c'est qu'il ne dépend pas d'une technologie services Web particulière (ex. REST ou SOAP), ce qui améliore son adaptabilité et son réutilisation. Pour faciliter le processus de sélection, une fonction de risque à base d'intégral floue à été proposée. Cette fonction est utilisée pour classer les compositions qui satisfont les exigences de vie privée. Cela permet de sélectionner la composition avec un minimum de risque de menace sur la vie privée. Enfin, et à fin de tester la validité et la faisabilité du modèle de vie privée et le Framework de sélection, nous avons proposé trois algorithmes de sélection. Le premier algorithme est de type meilleur d'abord. Dans cet algorithme, nous avons introduit une adaptation pour supporter les contraintes de vie privée. Les deux autres algorithmes se basent sur deux approches déclaratifs bien connues : la Satisfaisabilité booléenne (SAT), et le Answer

Set Programming (ASP). Pour ces deux approches, nous avons proposé un encodage pour résoudre le problème de Sélection des services Web avec Préservation de la vie Privée (PSWPP). L'efficacité de ces algorithmes a été testée et comparée sur différents types de jeux de données.

Dans la problématique de l'anonymat des données personnelles, nous avons proposé une approche, qui contrairement aux autres approches, n'introduit pas des modifications sur les données originales, mais utilise les techniques de Machine Learning pour construire des modèles à partir de ces données. Ces modèles sont utilisés par la suite pour générer des nouvelles données. Ces dernières diffères totalement des données originales, ce qui introduit une forte garantie de protection. L' étape de génération est guidée par un ensemble de règles sémantiques, ce qui augmente la qualité des données générées. Les résultats des tests concernant l'utilité des données générées sont très prometteurs et encouragent des éventuelles améliorations sur cette approche.

6.2 Perspectives

Comme tout travail, le présent travail est loin d'être complet. Les insuffisances de chaque approche sont déjà discutées et des perspectives à court terme sont introduites dans la conclusion de chaque chapitre. Nos perspectives à long terme se focalisent sur les points suivants :

— Vu la complexité et le vaste étendu de ce domaine, nous envisageons d'étendre notre étude à d'autres axes et applications telles-que : les réseaux sociaux, les systèmes de navigation et de géolocalisation, les différents types de réseaux mobiles et Ad hoc, etc.

— Explorer la protection de la vie privée dans d'autres environnements tel-que le cloud computing. La nature de cet environnement introduit plus de difficultés pour implémenter des solutions de protections de vie privée. Par exemple, la répartition des données dans le cloud dans divers sites pour optimiser le stockage, et le mouvement permanent de ces données entre sites à fin de réduire le temps de réponse, soulève de nouveaux problèmes comme la répartition des solutions, la difficulté des contrôles, et la complexité des contrats de l'utilisation des données. Le problème devient plus compliqué si les données sont réparties dans différents pays, soumis à des législations différentes et parfois non compatibles.

— Malgré les avancées dans les technologies de protection de la vie privée, le facteur humain reste le maillon faible de toute solution proposée dans ce domaine. Proposer des solutions à cette problématique dépasse de loin l'aspect technique, et s'étend à des aspects sociales et psychologiques. À notre

avis, explorer la possibilité de combiner les techniques informatiques (IA, Datamining, ...) avec les théories psychologiques et sociales, est un axe de recherche très motivant et très promoteur.

The grand aim of all science is to cover the greatest number of empirical facts by logical deduction from the smallest number of hypotheses or axioms.

Albert Einstein

A

Annexe A

▷

Contexte Technique et Définitions Préliminaires.

◁

Plan du chapitre

Cette section introduit quelques concepts de base utilisés tout au long du présent document. Elle définit la composition et la sélection des services Web, ainsi que certains concepts de base relatifs au WP-MaxSAT (WP-MaxSAT), le Answer Set Programming (ASP) et les règles d'association. Nous introduisons aussi quelques expérimentations complémentaires.

A.1 Contexte Technique et Définitions Préliminaires

A.1.1 Web Services Composition and Selection

Un service Web est défini comme une application modulaire et Auto-descriptive, invocable avec les technologies Web standards (HTTP, SOAP, etc.)[Curbera02, Sheng14]. Dans une architecture de service Web classique (voir Figure A.1), un fournisseur de services publie des descriptions de service sous la forme de fichiers WSDL (Web Service Description Language) sur des registres tels que UDDI (Universal Description Discovery and Integration) [Curbera02]. Les consommateurs peuvent découvrir les services du registre et enfin invoquer les services découverts. Pour certains types d'applications, il est nécessaire de combiner un ensemble de

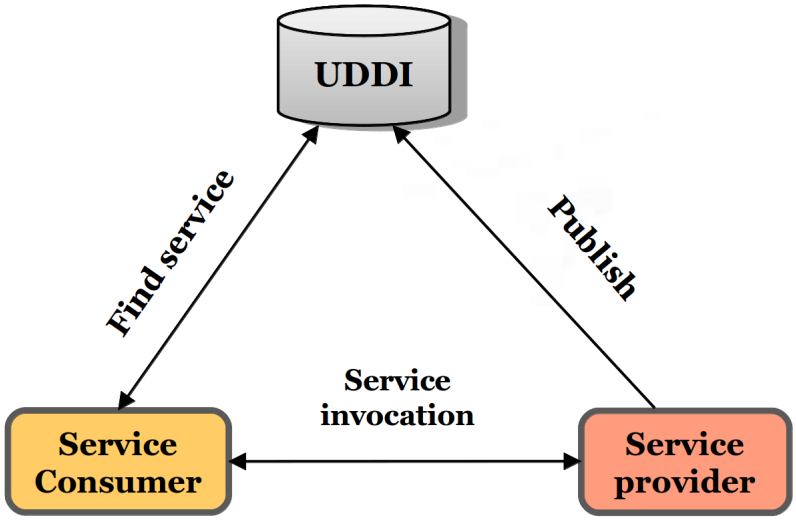


FIGURE A.1 – Architecture Web service.

services Web (services Web agrégés ou composites) pour répondre à des besoins plus complexes. Cette combinaison (composition ou Workflow) conduit à un service composite abstrait qui spécifie un ensemble de tâches atomiques ainsi que les flux de données et de contrôle entre ces tâches. Ce workflow peut être formalisé à l'aide de langages spécifiques tels que BPEL (Business Process Execution Language) [Jordan07].En tenant compte de la composition abstraite, la phase de sélection consiste à chercher un ensemble de services concrets répondant le mieux aux besoins non fonctionnels. En général, les exigences non fonctionnelles sont la

qualité de service (par exemple, la latence, la disponibilité, le coût, etc.), la sécurité ou, comme dans notre travail, les contraintes de vie privée.

A.1.2 SAT et MaxSAT partiel pondéré (Weighted Partial MaxSAT)

Une formule propositionnelle (ou booléenne) Φ est en forme normale conjonctive (CNF) s'il s'agit d'une conjonction ($\wedge$) de clauses, où une clause est une disjonction ($\vee$) de littéraux. Un littéral peut être une variable propositionnelle p ou sa négation $\neg p$. Une formule CNF peut être vue comme un ensemble de clauses et chaque clause comme un ensemble de littéraux. On note $Var(\Phi)$ l'ensemble des variables propositionnelles apparaissant dans Φ. Une affectation σ d'un ensemble $V \subseteq Var(\Phi)$ de variables propositionnelles est une appliquation $\sigma : V \rightarrow \{0, 1\}$

Une affectation σ satisfait une formule CNF Φ ($\sigma(\Phi) = 1$) si elle satisfait toutes ses clauses, dans ce cas, σ est appelé un modèle de Φ. Le problème de satisfiabilité propositionnelle (SAT) [Biere09] consiste à décider si une formule CNF donnée admet ou non un modèle. Aujourd'hui, ce problème NP-Complet a gagné une popularité considérable avec l'avènement d'une nouvelle génération de solveurs capables de résoudre de grandes instances CNF codant des problèmes du monde réel. En plus des applications traditionnelles de SAT, comme la vérification formelle de matérielle et logicielle, ces progrès impressionnants ont conduit à une utilisation croissante de la technologie SAT pour résoudre de nouvelles applications réelles telles que la planification, la bio-informatique, le Data-mining et la cryptographie. Dans la plupart de ces applications, nous nous intéressons principalement au problème de décision et à certaines de ses variantes d'optimisation telles que la satisfaction maximale (MaxSAT), le MaxSAT partiel (P-MaxSAT) ou le MaxSAT partiel pondéré (WP-MaxSAT).

MaxSAT est défini comme étant le problème de trouver une affectation de vérité satisfaisant un nombre maximum de clauses. Dans P-MaxSAT, étant donné une formule CNF $\Phi_h \wedge \Phi_s$, le problème est de trouver une affectation de vérité satisfaisant la partie dure (Φ_h), tout en maximisant le nombre de clauses souples satisfaites (Φ_s). Dans le problème MaxSAT partiel pondéré (WP-MaxSAT)[Li09a], la formule CNF contient également deux ensembles de clauses, un ensemble de clauses dures (Φ_h) qui doivent être satisfaites, et un ensemble de clauses logicielles pondérées (Φ_{ws}). Une solution à un problème WP-MaxSAT consiste à trouver une affectation optimale qui satisfait toutes les clauses dures (Φ_h) et maximise la somme des poids des clauses souples (Φ_{ws}) satisfaites.

A.1.3 Answer Set Programming (ASP)

Le Answer Set Programming (ASP) [Janhunen16, Kaufmann16], est devenue l'une des approches déclaratives les plus populaires pour résoudre des problèmes de recherche difficiles (NP-hard). L'ASP peut être considérée comme une branche de la représentation des connaissances, de la programmation logique et du raisonnement (non monotone). Syntaxiquement, ASP ressemble à la programmation logique (par exemple Prolog) alors que la sémantique est basée sur des modèles stables, appelés aussi ensembles de réponses (voir [Lifschitz16]). Un programme ASP Π est représenté comme un ensemble fini de règles. Une règle normale r_i est de la forme :

$$h_1 \vee \cdots \vee h_k \leftarrow a_1, ..., a_m, \neg a_{m+1}, ..., \neg a_n. \ \ (k \geqslant 0, n \geqslant m \geqslant 0)$$

Où h_i, a_j sont des atomes (littéraux) en langage de premier ordre, et $\neg$ représente un symbole de négation par échec. De plus, l'ensemble $H(r_i) = \{h_1, \ldots, h_k\}$ est appelée la tête de la règle r_i, et $B(r_i) = \{a_1, a_2, ..., a_m, \neg a_{m+1}, ..., \neg a_n\}$ est le corps de la règle r_i.

Si $B(r_i) = \emptyset$ la règle représente un fait ; alors que si $H(r_i) = \emptyset$, la règle représente une contrainte. Une solution pour un programme ASP correspond à son ensemble de réponses sous-jacent. Intuitivement, trouver l'ensemble de réponses est équivalent à trouver l'ensemble des seuls littérales qui sont justifiées par une règle dans le programme Π. L'utilisation de ASP implique la spécification du problème en tant que programme ASP, puis utiliser un solveur ASP pour traiter la spécification susmentionnée.

A.1.4 Data-mining et règles d'association

Le Data-mining est défini comme étant l'étude de la collecte, du nettoyage, du traitement, de l'analyse et de l'obtention d'informations utiles à partir des données. On général, les opérations de Data-mining se basent sur l'extraction des connaissances à partir des données transactionnelles. Ces dernières sont représentées souvent par une table transactionnelle qui se compose d'un ensemble de couples de type $(TID, Itemset)$, où TID est l'identificateur de transaction, et $Itemset$ est l'ensemble des items de la transaction (voir figure A.1). L'exemple typique d'une table

Transaction ID (TID)	Itemset
01	A,B,C
02	A,C
03	A,D
04	B,E,F

TABLE A.1 – Exemple d'une table transactionnelle.

transactionnelle est le panier de la ménagère, qui représente les enregistrements des clients et leurs achats.

A.1.4.1 Règles d'association

Soit $\mathcal{I} = \{i_1, i_2, \ldots, i_n\}$ un ensemble de n items. Soit $\mathcal{D} = \{T_1, T_2, \ldots, T_m\}$ un ensemble de transactions appelé aussi base de données. Chaque transaction T dans $\mathcal{D}$ a un identificateur de transaction TID unique et contient un sous-ensemble des éléments de $\mathcal{I}$ (itemset). Soit A un ensemble d'items. On dit qu'une transaction T contient A si $A \subseteq T$.

Une **règle d'association** est une implication de la forme $A \Rightarrow B$, où $A \subset \mathcal{I}$, $B \subset \mathcal{I}$, $A \neq \emptyset$, $B \neq \emptyset$ et $A \cap B = \emptyset$.

La règle $A \Rightarrow B$ tient dans l'ensemble de transactions D avec un **support** s, où s est le pourcentage de transactions dans D contenant $A \cup B$ (c'est-à-dire contenant A et B).

La règle $A \Rightarrow B$ a la **confiance** c dans l'ensemble de transactions D, où c est le pourcentage de transactions dans D contenant A qui contiennent aussi B.
Formellement, le support d'une règle d'association est défini comme suit :

$$support(A \Rightarrow B) = support(A \cup B) \tag{A.1}$$

où, le support d'un itemset $X \subset \mathcal{I}$ est défini comme suit :

$$support(X) = \frac{|\{T_i \subseteq \mathcal{D} \mid X \subseteq T_i\}|}{|\mathcal{D}|} \tag{A.2}$$

La confiance d'une règle d'association est défini comme suit :

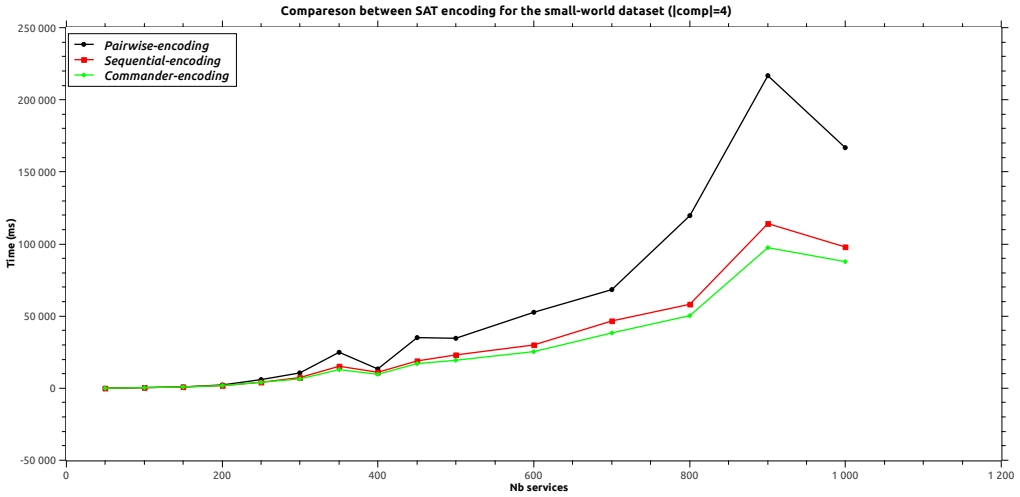
$$confiance(A \Rightarrow B) = \frac{support(A \cup B)}{support(A)} \tag{A.3}$$

A.2 Expérimentations complémentaires

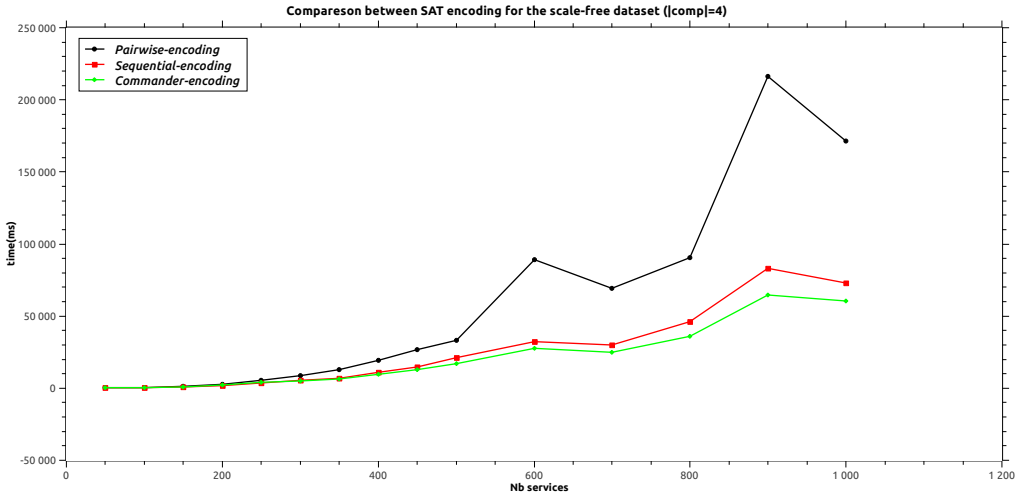
A.2.1 Justification de l'utilisation de l'encodage « Commander » (chapitre 4)

Nous avons évalué 03 schémas d'encodage existants : l'encodage «Pairwise», l'encodage « séquentiel » et l'encodage « Commander». L'expérimentation utilise les mêmes paramètres traités dans notre expérience de la section 4.6.5. À cette fin, nous avons fixé la taille de la composition à 4, et varié le nombre de services par

classe de 50 à 1000. Les résultats de cette expérience sont présentés dans la figure A.2.



(a)



(b)

FIGURE A.2 – Comparaison des encodages SAT : (a) Small-world dataset, (b) Scale-free dataset.

De cette expérimentation, nous pouvons clairement voir que le temps d'exécution est meilleur pour les instances qui ont utilisé l'encodage « Commander ».

A.2.2 Justification de l'utilisation du solveur Qmaxsat (chapitre 4)

Nous avons comparé l'efficacité de Qmaxsat avec deux solveurs : maxsatz2013f[1], un solveur « branch and bound » très efficace et CCLS_to_akmaxsat[2],

1. http://home.mis.u-picardie.fr/~cli/EnglishPage.html.
2. http://maxsat.ia.udl.cat/solvers/5/CCLS_to_akmaxsat_binaries.zip-201604080802.

le gagnant de la catégorie « Weighted Partial Max-SAT-random » dans Max-SAT Evaluation 2016[3]. Pour cette comparaison, nous avons utilisé quelques exemples de notre expérimentation de section 4.6.5. Les instances et les solveurs utilisés dans cette expérimentation peuvent être trouvés au lien suivant[4]. Notez que nous avons utilisé un « timeout » de 150 secondes. Les résultats de la comparaison sont décrits dans le tableau A.2.

	Time (second)		
Instances	**QMAXSAT**	**maxsatz2013f**	**CCLS_2_akmaxsat**
inst-sw-1000	46,43	Out of memory	timeout
inst-sw-900	96,29	Out of memory	timeout
inst-sw-800	42,22	timeout	timeout
inst-sw-700	141,948	timeout	timeout
inst-sw-600	51,356	timeout	timeout
inst-sw-500	13,116	31,49	timeout
inst-sw-450	6,06	18,98	timeout
inst-sw-400	6,1	12,8	timeout
inst-sw-350	14,152	21,48	timeout
inst-sw-300	3,316	7,052	140,53
inst-sw-250	2,604	4,42	43
inst-sw-200	0,66	2,80	12.32
inst-sw-150	0,596	1,584	7.52
inst-sw-100	0,356	0,780	2.39

TABLE A.2 – Comparaison entre les solveurs MAXSAT.

Les résultats montrent clairement que Qmaxsat offre les meilleures performances.

3. http://maxsat.ia.udl.cat
4. https://www.dropbox.com/s/20570m7hdbmvgk5/instances.zip?dl=0

Liste des publications

Journaux internationaux avec comité de lecture

— Amine Belabed, Esma Aimeur, Mohammed Amine Chikh & Fethallah Hadjila. "A Privacy-Preserving Approach for Composite Web Service Selection". Transactions on Data Privacy, vol. 10, no. 2, pages 83-115, INST ESTUDIOS DOCUMENTALES CIENCIA & TECNOLOGIAIEDCYT JOAQUIN COSTA 22, MADRID, 28002, SPAIN, 2017.

Conférences internationales avec comité de lecture

— Amine Belabed, Esma Aïmeur & Amine Chikh. "A personalized whitelist approach for phishing webpage detection". In Seventh International Conference on Availability, Reliability and Security (ARES), 2012, pages 249-254. IEEE, 2012.

— Amine Belabed, Amine Chikh & Esma Aïmeur. "Une approche à base de Machine Learning pour la protection des micro-données". In Colloque sur l'optimisation et les systèmes d'information, COSI 2014, Lamos, pages 206-214. COSI, 2014.

— Wakrime, Abderrahim Ait, Said Jabbour, and Amine Belabed. "Web Service Composition as minimal unsatisfiability." Electrical and Information Technologies (ICEIT), 2016 International Conference on. IEEE, 2016.

Conférences nationales avec comité de lecture

— Naziha Abderahim, Asma Bensidhoum & Amine Belabed. "Un agent P3P pour la protection de la vie privée des utilisateurs Web". JEESI, Alger, 2012.

Bibliographie

[Abderahim12] Naziha Abderahim, Asma Bensidhoum & Amine Belabed. *Un agent P3P pour la protection de la vie privée des utilisateurs Web*. JEESI, 2012. 29

[Abowd01] John M Abowd & Simon D Woodcock. *Disclosure limitation in longitudinal linked data*. Confidentiality, Disclosure, and Data Access : Theory and Practical Applications for Statistical Agencies, vol. 215277, North Holland, 2001. 102

[Abu-Nimeh07] Saeed Abu-Nimeh, Dario Nappa, Xinlei Wang & Suku Nair. *A comparison of machine learning techniques for phishing detection*. In Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit, pages 60–69. ACM, 2007. 38

[Ado16] Kukic Ado. The definitive guide to single sign on. Auth0, 2016. 23

[Aggarwal08] Charu C Aggarwal & Philip S Yu. *On static and dynamic methods for condensation-based privacy-preserving data mining*. ACM Transactions on Database Systems (TODS), vol. 33, no. 1, page 2, ACM, 2008. 101

[Aggarwal16] Charu C Aggarwal*et al.* Recommender systems. Springer, 2016. 18

[Aguiar07] Rui L Aguiar, Amardeo Sarma, Dennis Bijwaard, Loris Marchetti & Piotr Pacyna. *Pervasiveness in a competitive multi-operator environment : the daidalos project*. IEEE Communications Magazine, vol. 45, no. 10, IEEE, 2007. 23

[Ahvanooey17] Milad Taleby Ahvanooey, Qianmu Li, Mahdi Rabbani & Ahmed Raza Rajput. *A Survey on Smartphones Security : Software Vulnerabilities, Malware, and Attacks*. INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, vol. 8, no. 10, pages 30–45, SCIENCE & INFORMATION, ENGLAND, 2017. 19

[Alliance02] Liberty Alliance. *Liberty alliance project*. Web page at http ://www. projectliberty. org, 2002. 24

[Alrifai12] Mohammad Alrifai, Thomas Risse & Wolfgang Nejdl. *A hybrid approach for efficient Web service composition with end-to-end QoS constraints*. ACM Transactions on the Web (TWEB), vol. 6, no. 2, pages 7 :1–7 :31, ACM, 2012. 52

[Anderson04] Anne H Anderson. *The Relationship Between XACML and P3P Privacy Policies*. Sun Microsystems, vol. 11, 2004. 31

[Ashley03] Paul Ashley, Satoshi Hada, Günter Karjoth, Calvin Powers & Matthias Schunter. *Enterprise privacy authorization language (EPAL)*. IBM Research, 2003. 30, 31

[Aza11] Raskin Aza. *Tabnabbing : A New Type of Phishing Attack*. `http://www.azarask.in/blog/post/a-new-type-of-phishing-attack/`, 2011. Accessed : 2017-05-14. 37

[Bai16] Jinrong Bai & Junfeng Wang. *Improving malware detection using multi-view ensemble learning*. Security and Communication Networks, vol. 9, no. 17, pages 4227–4241, Wiley Online Library, 2016. 19

[Barrett09] Clark W Barrett, Roberto Sebastiani, Sanjit A Seshia & Cesare Tinelli. *Satisfiability Modulo Theories*. Handbook of satisfiability, vol. 185, pages 825–885, 2009. 91

[Barth11] Adam Barth. *RFC 6265 : HTTP state management mechanism*. 2011. 19

[Bayardo05] Roberto J Bayardo & Rakesh Agrawal. *Data privacy through optimal k-anonymization*. In Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on, pages 217–228. IEEE, 2005. 99

[Belabed12] Amine Belabed, Esma Aïmeur & Amine Chikh. *A personalized whitelist approach for phishing webpage detection*. In Seventh International Conference on Availability, Reliability and Security (ARES), 2012, pages 249–254. IEEE, 2012. 6

[Belabed14] Amine Belabed, Amine Chikh & Esma Aïmeur. *Une approche à base de Machine Learning pour la protection des micro-données*. In Colloque sur l'optimisation et les systèmes d'information, COSI 2014, Lamos, pages 206–214. COSI, 2014. 7

[Belabed17] Amine Belabed, Esma Aimeur, Mohammed Amine Chikh & Fethallah Hadjila. *A Privacy-Preserving Approach for Composite Web Service Selection*. TRANSACTIONS ON DATA PRIVACY, vol. 10, no. 2, pages 83–115, INST ESTUDIOS DOCUMENTALES CIENCIA & TECNOLOGIA-IEDCYT JOAQUIN COSTA 22, MADRID, 28002, SPAIN, 2017. 6

[Bergholz08] Andre Bergholz, Jeong Ho Chang, Gerhard Paass, Frank Reichartz & Siehyun Strobel. *Improved Phishing Detection using Model-Based Features.* In CEAS, 2008. 36

[Bernsmed12] Karin Bernsmed, Åsmund Ahlmann Nyre & Martin Gilje Jaatun. *User Agents for Matching Privacy Policies with User Preferences*. International Journal of Computer Theory and Engineering, vol. 4, no. 3, page 451, IACSIT Press, 2012. 29

[Bhowmick18] Alexy Bhowmick & Shyamanta M Hazarika. *E-Mail Spam Filtering : A Review of Techniques and Trends*. In Advances in Electronics, Communication and Computing, pages 583–590. Springer, 2018. 18

[Biere09] Armin Biere, Marijn Heule & Hans van Maaren. Handbook of satisfiability, volume 185. IOS press, 2009. 7, 53, 124

[Bok89] Sissela Bok. Secrets : On the ethics of concealment and revelation. Vintage, 1989. 12

[Bollobás03] Béla Bollobás, Christian Borgs, Jennifer Chayes & Oliver Riordan. *Directed scale-free graphs*. In Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms, pages 132–139. Society for Industrial and Applied Mathematics, 2003. 80

[Bradley97] Andrew P Bradley. *The use of the area under the ROC curve in the evaluation of machine learning algorithms*. Pattern recognition, vol. 30, no. 7, pages 1145–1159, Elsevier, 1997. 109

[Brands00] Stefan A Brands. Rethinking public key infrastructures and digital certificates : building in privacy. Mit Press, 2000. 3

[Broomhead88] David S Broomhead & David Lowe. *Radial basis functions, multi-variable functional interpolation and adaptive networks*. Rapport technique, Royal Signals and Radar Establishment Malvern (United Kingdom), 1988. 109

[Bujlow17] Tomasz Bujlow, Valentín Carela-Español, Josep Sole-Pareta & Pere Barlet-Ros. *A survey on web tracking : Mechanisms,*

implications, and defenses. Proceedings of the IEEE, vol. 105, no. 8, pages 1476–1510, IEEE, 2017. 18

[Burridge03] Jim Burridge. *Information preserving statistical obfuscation*. Statistics and Computing, vol. 13, no. 4, pages 321–327, Springer, 2003. 102

[Camenisch01] Jan Camenisch & Anna Lysyanskaya. *An efficient system for non-transferable anonymous credentials with optional anonymity revocation*. In International Conference on the Theory and Applications of Cryptographic Techniques, pages 93–118. Springer, 2001. 24

[Camenisch02] Jan Camenisch & Els Van Herreweghen. *Design and implementation of the idemix anonymous credential system*. In Proceedings of the 9th ACM conference on Computer and communications security, pages 21–30. ACM, 2002. 24

[Cao08] Ye Cao, Weili Han & Yueran Le. *Anti-phishing based on automated individual white-list*. In Proceedings of the 4th ACM workshop on Digital identity management, pages 51–60. ACM, 2008. 35, 37, 49

[Carminati15] Barbara Carminati, Elena Ferrari & Ngoc Hong Tran. *A privacy-preserving framework for constrained choreographed service composition*. In IEEE International Conference on Web Services (ICWS), pages 297–304. IEEE, 2015. 52, 56

[Chang11] Chih-Chung Chang & Chih-Jen Lin. *LIBSVM : a library for support vector machines*. ACM transactions on intelligent systems and technology (TIST), vol. 2, no. 3, page 27, Acm, 2011. 46, 108

[Charu08] A Charu & S Yu Philip. *Privacy-preserving data mining : Models and algorithms*. ASPVU, Boston, 2008. 99

[Chaum81] David L Chaum. *Untraceable electronic mail, return addresses, and digital pseudonyms*. Communications of the ACM, vol. 24, no. 2, pages 84–90, ACM, 1981. 27

[Chaum83] David Chaum. *Blind signatures for untraceable payments*. In Advances in cryptology, pages 199–203. Springer, 1983. 24

[Chaum85] David Chaum. *Security without identification : Transaction systems to make big brother obsolete*. Communications of the ACM, vol. 28, no. 10, pages 1030–1044, ACM, 1985. 23, 24

[Chaum91] David Chaum & Eugène Van Heyst. *Group signatures*. In Workshop on the Theory and Application of of Cryptographic Techniques, pages 257–265. Springer, 1991. 24, 25

[Chen09] Bee-Chung Chen, Daniel Kifer, Kristen LeFevre & Ashwin Machanavajjhala. *Privacy-Preserving Data Publishing*. Found. Trends databases, vol. 2, no. 1–2, pages 1–167, Now Publishers Inc., Hanover, MA, USA, 2009. 21

[Chen15] Jilin Chen, Eben M Haber, Ruogu Kang, Gary Hsieh & Jalal Mahmud. *Making Use of Derived Personality : The Case of Social Media Ad Targeting.* In ICWSM, pages 51–60, 2015. 18

[Cherifi10] Chantal Cherifi, Vincent Labatut & Jean-François Santucci. *Benefits of semantics on web service composition from a complex network perspective*. In International Conference on Networked Digital Technologies, pages 80–90. Springer, 2010. 80

[Cingolani12] Pablo Cingolani & Jesus Alcala-Fdez. *jFuzzyLogic : a robust and flexible Fuzzy-Logic inference system language implementation.* In FUZZ-IEEE, pages 1–8. Citeseer, 2012. 73

[Ciriani07] V Ciriani, SSF De Capitani di Vimercati & P Samarati. *k-Anonymity. Secure Data Management in Decentralized Systems. 2007*, 2007. 94

[Clement08] Andrew Clement, David Ley, Terry Costantino, Dan Kurtz & Mike Tissenbaum. *The PIPWatch Toolbar : Combining PIPEDA, PETs and market forces through social navigation to enhance privacy protection and compliance*. In Technology and Society, 2008. ISTAS 2008. IEEE International Symposium on, pages 1–10. IEEE, 2008. 29

[cni16] *La protection des données dans le monde*. `https://www.cnil.fr/fr/la-protection-des-donnees-dans-le-monde`, 2016. Accessed : 2018-01-10. vii, 13

[Costa17] Núria Costa. *Mixnets for long-term privacy*. Second In, page 410, 2017. 27

[Costante13] Elisa Costante, Federica Paci & Nicola Zannone. *Privacy-aware web service composition and ranking*. In 20th IEEE International Conference on Web Services (ICWS), pages 131–138. IEEE, 2013. 52, 54

[Cranor02a] Lorrie Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall & Joseph Reagle. *The platform for privacy preferences 1.0 (P3P1. 0) specification*. W3C recommendation, vol. 16, 2002. 7, 30, 53

[Cranor02b] Lorrie Faith Cranor, Manjula Arjula & Praveen Guduru. *Use of a P3P user agent by early adopters*. In Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society, pages 1–10. ACM, 2002. 29

[CTRL18] Cyberoam Threat Research Labs CTRL. *The Phishing menace and ways to protect your online identity*. `https://www.cyberoam.com/phishing.html`, 2018. Accessed : 2018-01-10. 5

[Curbera02] Francisco Curbera, Matthew Duftler, Rania Khalaf, William Nagy, Nirmal Mukhi & Sanjiva Weerawarana. *Unraveling the Web services web : an introduction to SOAP, WSDL, and UDDI*. IEEE Internet computing, vol. 6, no. 2, pages 86–93, IEEE, 2002. 123

[Dake05] Dake. *diagramm : Zero Knowledge Interactive Proof - example with Ali Baba's cave*. `https://commons.wikimedia.org/wiki/File:Zkip_alibaba1.png`, 2005. vii, 26

[Dari Bekara12] Kheira Dari Bekara. *Protection des données personnelles côté utilisateur dans le E-Commerce*. Doctorat, Evry, Institut national des télécommunications, 2012. 31

[DeCew97] Judith Wagner DeCew. In pursuit of privacy : Law, ethics, and the rise of technology. Cornell University Press, 1997. 12

[DeCew06] Judith Wagner DeCew. *Privacy and policy for genetic research*. Ethics, Computing, and Genomics. Jones and Bartlett, Sudbury, MA, pages 121–136, 2006. 12

[Dewri08] Rinku Dewri, Indrajit Ray, Indrakshi Ray & Darrell Whitley. *On the Optimal Selection of k in the k-Anonymity Problem*. In Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on, pages 1364–1366. IEEE, 2008. 100

[Diffie76] Whitfield Diffie & Martin Hellman. *New directions in cryptography*. IEEE transactions on Information Theory, vol. 22, no. 6, pages 644–654, IEEE, 1976. 24, 25

[Dingledine04] Roger Dingledine, Nick Mathewson & Paul Syverson. *Tor : The second-generation onion router*. Rapport technique, Naval Research Lab Washington DC, 2004. 27

[dir08] *EU Data Protection Directive (Directive 95/46/EC).* http://whatis.techtarget.com/definition/EU-Data-Protection-Directive-Directive-95-46-EC, 2008. Accessed : 2018-01-20. 15, 16

[Directive95] EU Directive. *95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data.* Official Journal of the EC, vol. 23, no. 6, 1995. 14, 15

[Domingo01] Josep Domingo Ferrer & Vicenc Torra. *Disclosure control methods and information loss for microdata.* Confidentiality, disclosure, and data access : theory and practical applications for statistical agencies, pages 91–110, Citeseer, 2001. 99

[Domingo08] Josep Domingo Ferrer. *A survey of inference control methods for privacy-preserving data mining.* In Privacy-preserving data mining, pages 53–80. Springer, 2008. 98

[Duncan00] George T Duncan & Sumitra Mukherjee. *Optimal disclosure limitation strategy in statistical databases : Deterring tracker attacks through additive noise.* Journal of the American Statistical Association, vol. 95, no. 451, pages 720–729, Taylor & Francis Group, 2000. 98

[Dwork11] Cynthia Dwork, Frank McSherry, Kobbi Nissim & Adam Smith. *Differential privacyâa primer for the perplexed,â.* Joint UNECE/Eurostat work session on statistical data confidentiality, vol. 11, 2011. 94, 101

[Efron92] Bradley Efron. *Bootstrap methods : another look at the jackknife.* In Breakthroughs in statistics, pages 569–593. Springer, 1992. 102

[Emigh07] Aaron Emigh. *Phishing attacks : Information flow and chokepoints.* Phishing and Countermeasures, pages 31–64, Wiley, 2007. 36

[Fawcett06] Tom Fawcett. *An introduction to ROC analysis.* Pattern recognition letters, vol. 27, no. 8, pages 861–874, Elsevier, 2006. 109

[Feige88] Uriel Feige, Amos Fiat & Adi Shamir. *Zero-knowledge proofs of identity.* Journal of cryptology, vol. 1, no. 2, pages 77–94, Springer, 1988. 26

[Feng15] Zhiyong Feng, Bo Lan, Zhen Zhang & Shizhan Chen. *A study of semantic web services network*. The Computer Journal, vol. 58, no. 6, pages 1293–1305, Br Computer Soc, 2015. 80

[Fette07] Ian Fette, Norman Sadeh & Anthony Tomasic. *Learning to detect phishing emails*. In Proceedings of the 16th international conference on World Wide Web, pages 649–656. ACM, 2007. 38

[Fienberg98] Stephen E Fienberg & Russell J Steele. *Disclosure limitation using perturbation and related methods for categorical data*. Journal of Official Statistics, vol. 14, no. 4, page 485, Statistics Sweden (SCB), 1998. 102

[Fournier12] Philippe Fournier Viger, Cheng-Wei Wu & Vincent S Tseng. *Mining top-k association rules*. In Canadian Conference on Artificial Intelligence, pages 61–73. Springer, 2012. 113

[Fournier16] Philippe Fournier Viger, Jerry Chun-Wei Lin, Antonio Gomariz, Ted Gueniche, Azadeh Soltani, Zhihong Deng & Hoang Thanh Lam. *The SPMF open-source data mining library version 2*. In Joint European conference on machine learning and knowledge discovery in databases, pages 36–40. Springer, 2016. 113

[Gambs10] Sébastien Gambs, Marc-Olivier Killijian & Miguel Núñez del Prado Cortez. *Show me how you move and I will tell you who you are*. In Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS, pages 34–41. ACM, 2010. 22

[Gambs12] Sébastien Gambs. *Inference attacks on geolocated data*. Cybercrime, page 23, 2012. 22

[Gambs14] Sébastien Gambs, Marc-Olivier Killijian & Miguel Núñez del Prado Cortez. *De-anonymization attack on geolocated data*. Journal of Computer and System Sciences, vol. 80, no. 8, pages 1597–1614, Elsevier, 2014. 22

[Garera07] Sujata Garera, Niels Provos, Monica Chew & Aviel D Rubin. *A framework for detection and measurement of phishing attacks*. In Proceedings of the 2007 ACM workshop on Recurring malcode, pages 1–8. ACM, 2007. 39

[Gebser14] Martin Gebser, Roland Kaminski, Benjamin Kaufmann & Torsten Schaub. *Clingo= ASP+ control : Preliminary report*. arXiv preprint arXiv :1405.3694, 2014. 80

[Gerlach15] Jin Gerlach, Thomas Widjaja & Peter Buxmann. *Handle with care : How online social network providersâ privacy policies impact usersâ information sharing behavior*. The Journal of Strategic Information Systems, vol. 24, no. 1, pages 33–43, Elsevier, 2015. 11

[Ghazinour11] Kambiz Ghazinour & Ken Barker. *Capturing P3P semantics using an enforceable lattice-based structure*. In Proceedings of the 4th International Workshop on Privacy and Anonymity in the Information Society, pages 4 :1–4 :6. ACM, 2011. 7, 53, 62

[Ghazinour14] Kambiz Ghazinour, Amir H Razavi & Ken Barker. *A Model for Privacy Compromisation Value*. Procedia Computer Science, vol. 37, pages 143–152, Elsevier, 2014. 62

[Ginosar17] Avshalom Ginosar & Yaron Ariel. *An analytical framework for online privacy research : What is missing ?* Information & Management, vol. 54, no. 7, pages 948–957, Elsevier, 2017. 11

[Golawasser85] S Golawasser, S Michli & C Rackoff. *The knowledge complexity of interactive proofs system*. In Proceedings of the 17th ACM Symposium on Theory of Computing, Providence, Rhode Island, United States, pages 291–314, 1985. 26

[Goldenberg14] Meir Goldenberg, Ariel Felner, Roni Stern, Guni Sharon, Nathan Sturtevant, Robert C Holte & Jonathan Schaeffer. *Enhanced partial expansion A**. Journal of Artificial Intelligence Research, vol. 50, no. 1, pages 141–187, AI Access Foundation, 2014. 74

[Goldreich94] Oded Goldreich & Yair Oren. *Definitions and properties of zero-knowledge proof systems*. Journal of Cryptology, vol. 7, no. 1, pages 1–32, Springer, 1994. 26

[Goldreich96] Oded Goldreich & Ariel Kahan. *How to construct constant-round zero-knowledge proof systems for NP*. Journal of Cryptology, vol. 9, no. 3, pages 167–189, Springer, 1996. 26

[Google12] Developers Google. *Safe Browsing, API.*, 2012. 35, 37

[Grabisch96] Michel Grabisch. *The application of fuzzy integrals in multicriteria decision making*. European journal of operational research, vol. 89, no. 3, pages 445–456, Elsevier, 1996. 7, 53, 71

[Grabisch08] Michel Grabisch, Ivan Kojadinovic & Patrick Meyer. *A review of methods for capacity identification in Choquet integral*

based multi-attribute utility theory : Applications of the Kappalab R package. European journal of operational research, vol. 186, no. 2, pages 766–785, Elsevier, 2008. 72, 73

[Group17] Anti-Phishing Working Group*et al.* *Phishing activity trends report*. Anti-Phishing Working Group, 2017. 20, 36

[Guermouche07] Nawal Guermouche, Salima Benbernou, Emmanuel Coquery & Mohand-Said Hacid. *Privacy-aware web service protocol replaceability*. In IEEE International Conference on Web Services, ICWS., pages 1048–1055. IEEE, 2007. 52, 55

[Gunn04] Patrick P Gunn, Allen M Fremont, Melissa Bottrell, Lisa R Shugarman, Jolene Galegher & Tora Bikson. *The health insurance portability and accountability act privacy rule : a practical guide for researchers*. Medical care, vol. 42, no. 4, pages 321–327, LWW, 2004. 16

[Gupta17] Shashank Gupta & Brij Bhooshan Gupta. *Cross-Site Scripting (XSS) attacks and defense mechanisms : classification and state-of-the-art*. International Journal of System Assurance Engineering and Management, vol. 8, no. 1, pages 512–530, Springer, 2017. 20

[Hagberg13] Aric Hagberg, Dan Schult, Pieter Swart, D Conway, L Séguin-Charbonneau, C Ellison, B Edwards & J Torrents. *Networkx. High productivity software for complex networks*. Webová strá nka https ://networkx. lanl. gov/wiki, 2013. 80

[Halfaoui15] Amal Halfaoui, Hadjila Fethallah & Fedoua Didi. *QoS-Aware Web Services Selection Based on Fuzzy Dominance*. In Computer Science and Its Applications - 5th IFIP TC 5 International Conference, CIIA 2015, Saida, Algeria, May 20-21, 2015, Proceedings, pages 291–300, 2015. 52

[Hansen08] Marit Hansen, Ari Schwartz & Alissa Cooper. *Privacy and identity management*. IEEE Security & Privacy, vol. 6, no. 2, IEEE, 2008. 22

[Hart68] Peter E Hart, Nils J Nilsson & Bertram Raphael. *A formal basis for the heuristic determination of minimum cost paths*. IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pages 100–107, IEEE, 1968. 73

[Hayat07] Muhammad Aslam Hayat. *Privacy and Islam : From the Quran to data protection in Pakistan*. Information & Communications Technology Law, vol. 16, no. 2, pages 137–148, Taylor & Francis, 2007. 13

[He11] Mingxing He, Shi-Jinn Horng, Pingzhi Fan, Muhammad Khurram Khan, Ray-Shine Run, Jui-Lin Lai, Rong-Jian Chen & Adi Sutanto. *An efficient phishing webpage detector*. Expert Systems with Applications, vol. 38, no. 10, pages 12018–12027, Elsevier, 2011. 39, 43, 47

[Heer93] GR Heer. *A bootstrap procedure to preserve statistical confidentiality in contingency tables*. In Proceedings of the international seminar on statistical confidentiality, pages 261–271. Luxembourg : Office for official publications of the european communities, 1993. 98

[Hong18] Jiwon Hong, Sanghyun Park, Sang-Wook Kim, Dongphil Kim & Wonho Kim. *Classifying malwares for identification of author groups*. Concurrency and Computation : Practice and Experience, Wiley Online Library, 2018. 19

[Huang09] Angus FM Huang, Ci-Wei Lan & Stephen JH Yang. *An optimal QoS-based Web service selection scheme*. Information Sciences, vol. 179, no. 19, pages 3309–3322, Elsevier, 2009. 52

[Huang12] Keman Huang, Yushun Fan & Wei Tan. *An empirical study of programmable web : A network analysis on a service-mashup system*. In Web Services (ICWS), 2012 IEEE 19th International Conference on, pages 552–559. IEEE, 2012. 80, 83

[Inness96] Julie C Inness. Privacy, intimacy, and isolation. Oxford University Press on Demand, 1996. 12

[ins08] *In Session Phishing Attacks Trusteer Research Paper*. `www.trusteer.com`, 2008. Accessed : 2018-02-11. 37

[Iyengar02] Vijay S Iyengar. *Transforming data to satisfy privacy constraints*. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 279–288. ACM, 2002. 99

[Jakob13] Kallin Jakob & Lobo Valbuena Irene. *A comprehensive tutorial on cross-site scripting*. `https://excess-xss.com/`, 2013. Accessed : 2018-01-25. 20

[Janhunen16] Tomi Janhunen & Ilkka Nimelä. *The answer set programming paradigm*. AI Magazine, vol. 37, no. 3, pages 13–24, 2016. 7, 53, 125

[Jordan07] Diane Jordan, John Evdemon, Alexandre Alves, Assaf Arkin, Sid Askary, Charlton Barreto, Ben Bloch, Francisco Curbera,

Mark Ford, Yaron Goland*et al.* *Web services business process execution language version 2.0*. OASIS standard, vol. 11, no. 120, page 5, 2007. 123

[Jøsang07] Audun Jøsang, Muhammed Al Zomai & Suriadi Suriadi. *Usability and privacy in identity management architectures*. In Proceedings of the fifth Australasian symposium on ACSW frontiers-Volume 68, pages 143–152. Australian Computer Society, Inc., 2007. 23

[jun17] *Juniper : Détails techniques sur WannaCry.* `https://www.informatiquenews.fr/juniper-details-techniques-wannacry-51980`, 2017. Accessed : 2018-01-25. 19

[Kaufmann16] Benjamin Kaufmann, Nicola Leone, Simona Perri & Torsten Schaub. *Grounding and solving in answer set programming*. AI Magazine, vol. 37, no. 3, pages 25–32, 2016. 125

[Ke13] Changbo Ke, Zhiqiu Huang & Mei Tang. *Supporting negotiation mechanism privacy authority method in cloud computing*. Knowledge-Based Systems, vol. 51, pages 48–59, Elsevier, 2013. 52, 54

[Ke15] Changbo Ke, Ruchuan Wang, Fu Xiao & Zhiqiu Huang. *Requirement-Oriented Privacy Protection Analysis Architecture in Cloud Computing*. Journal of Communications, vol. 10, no. 1, pages 55–63, 2015. 52, 54

[Khonji11] AJ Mahmoud Khonji & Youssef Iraqi. *A Brief Description of 47 Phishing Classification Features*. 2011. 38

[Kil09] Hyunyoung Kil, Seog-Chan Oh, Ergin Elmacioglu, Wonhong Nam & Dongwon Lee. *Graph theoretic topological analysis of web service networks*. World Wide Web, vol. 12, no. 3, pages 321–343, Springer, 2009. 80

[Klieber07] Will Klieber & Gihwon Kwon. *Efficient CNF encoding for selecting 1 from N objects*. In Proc. International Workshop on Constraints in Formal Verification, 2007. 76, 77

[Kohavi95] Ron Kohavi. *The power of decision tables*. In European conference on machine learning, pages 174–189. Springer, 1995. 109

[Koshimura12] Miyuki Koshimura, Tong Zhang, Hiroshi Fujita & Ryuzo Hasegawa. *Qmaxsat : A partial max-sat solver*. Journal on Satisfiability, Boolean Modeling and Computation, vol. 8, pages 95–100, 2012. 80

[Kwon11] Ohbyung Kwon, Yonnim Lee & Debashis Sarangib. *A Galois lattice approach to a context-aware privacy negotiation service*. Expert Systems with Applications, vol. 38, no. 10, pages 12619–12629, Elsevier, 2011. 52, 55

[Langheinrich02] Marc Langheinrich, Lorrie Cranor & Massimo Marchiori. *Appel : A p3p preference exchange language*. W3C Working Draft, 2002. 30

[Léonard99] Thierry Léonard. *La protection des données à caractère personnel en pleine (r) evolution-La loi du 11 décembre 1998 transposant la directive 95/46/CE du 24 octobre 1995*. Journal des Tribunaux, vol. 1, page 377, 1999. 15

[Levenshtein65] Vladimir I Levenshtein. *Binary codes capable of correcting deletions, insertions and reversals*. Doklady. Akademii Nauk SSSR, vol. 163, no. 4, pages 845–848, 1965. 38

[Levy05] Stephen E Levy & Carl Gutwin. *Improving understanding of website privacy policies with fine-grained policy anchors*. In Proceedings of the 14th international conference on World Wide Web, pages 480–488. ACM, 2005. 29

[Li07] Ninghui Li, Tiancheng Li & Suresh Venkatasubramanian. *t-closeness : Privacy beyond k-anonymity and l-diversity*. In Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on, pages 106–115. IEEE, 2007. 101

[Li09a] Chu Min Li & Felip Manyà. *Maxsat*. In Handbook of satisfiability, chapitre 19, pages 613–631. ios press, 2009. 124

[Li09b] Chu Min Li, Felip Manyà, Nouredine Mohamedou & Jordi Planes. *Exploiting cycle structures in Max-SAT*. In International Conference on Theory and Applications of Satisfiability Testing, pages 467–480. Springer, 2009. 80

[Li14] Xianxian Li*et al.* *Personalized privacy protection for transactional data*. In International Conference on Advanced Data Mining and Applications, pages 253–266. Springer, 2014. 22

[Lifschitz02] Vladimir Lifschitz. *Answer set programming and plan generation*. Artificial Intelligence, vol. 138, no. 1, pages 39–54, Elsevier, 2002. 78

[Lifschitz16] Vladimir Lifschitz. *Answer sets and the language of answer set programming*. AI Magazine, vol. 37, no. 3, pages 7–12, 2016. 125

[Liu13] YC Liu & YB Liu. *A sort of web service selection strategy based on the fusion of QoS and service reliability*. International

Journal of Computer Science Issues, vol. 10, no. 1, pages 414–420, 2013. 52

[Ltd17] Check Point Software Technologies Ltd. *Whitepaper :An in-depth analysis of the copycat android malware campaign.* `https://www.checkpoint.com/downloads/resources/copycat-research-report.pdf`, 2017. Accessed : 2018-01-25. 19

[Lunacek06] Monte Lunacek, Darrell Whitley & Indrakshi Ray. *A crossover operator for the k-anonymity problem*. In Proceedings of the 8th annual conference on Genetic and evolutionary computation, pages 1713–1720. ACM, 2006. 99

[Luo15] Chuan Luo, Shaowei Cai, Wei Wu, Zhong Jie & Kaile Su. *CCLS : an efficient local search algorithm for weighted maximum satisfiability*. IEEE Transactions on Computers, vol. 64, no. 7, pages 1830–1843, IEEE, 2015. 80

[Ma09] Justin Ma, Lawrence K Saul, Stefan Savage & Geoffrey M Voelker. *Beyond blacklists : learning to detect malicious web sites from suspicious URLs*. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 1245–1254. ACM, 2009. 39

[Machanavajjhala07] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke & Muthuramakrishnan Venkitasubramaniam. *L-diversity : Privacy beyond k-anonymity*. ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, no. 1, page 3, ACM, 2007. 100

[Marichal00] Jean-Luc Marichal. *An axiomatic approach of the discrete Choquet integral as a tool to aggregate interacting criteria*. IEEE Transactions on Fuzzy Systems, vol. 8, no. 6, pages 800–807, IEEE, 2000. 71, 73

[McNeil84] Barbara J McNeil & James A Hanley. *Statistical approaches to the analysis of receiver operating characteristic (ROC) curves*. Medical decision making, vol. 4, no. 2, pages 137–150, Sage Publications Sage CA : Thousand Oaks, CA, 1984. 109

[Mendel13] Toby Mendel, Andrew Puddephatt, Ben Wagner, Dixie Hawtin & Natalia Torres. *LE RESPECT DE LA VIE PRIVÉE SUR L'INTERNET ET LA LIBERTÉ D'EXPRESSION*. Unesco, 2013. 11

[Meyerson04] Adam Meyerson & Ryan Williams. *On the complexity of optimal k-anonymity*. In Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pages 223–228. ACM, 2004. 94, 99

[mic18] *Microsoft Account*. `https://account.microsoft.com/account`, 2018. Accessed : 2018-02-15. 23

[Mittal17] Sandeep Mittal & Priyanka Sharma. *General Data Protection Regulation (GDPR)*. Asian Journal of Computer Science And Information Technology, vol. 7, no. 4, 2017. 15

[Moore10] Adam D Moore. *Privacy, public health, and controlling medical information*. In HEC forum, volume 22, pages 225–240. Springer, 2010. 11, 12

[Moore13] Adam D. Moore. Privacy. Blackwell Publishing Ltd, 2013. 11, 12

[Morin08] Dave Morin. *Announcing facebook connect*. [online]. Facebook, May, vol. 9, 2008. 23

[Morton12] Stuart Morton, Malika Mahoui & P Joseph Gibson. *An automated data utility clustering methodology using data constraint rules*. In Proceedings of the 2012 international workshop on Smart health and wellbeing, pages 9–16. ACM, 2012. 100

[moz17] *HTTP cookies*. `https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies`, 2017. Accessed : 2018-01-25. 20

[moz18] *How does built-in Phishing and Malware Protection work?* `https://support.mozilla.org/en-US/kb/how-does-phishing-and-malware-protection-work`, 2018. Accessed : 2018-01-10. 35, 37

[Nadimpalli11] Sandeep Varma Nadimpalli & Valli Kumari Vatsavayi. *BM (Break-Merge) : An Elegant Approach for Privacy Preserving Data Publishing*. In Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on, pages 1202–1207. IEEE, 2011. 100

[Neven08] Gregory Neven. *A quick introduction to anonymous credentials, September 2008*. `https://idemix.files.wordpress.com/2009/08/neven2008-quick_introduction_to_anonymous_credentials.pdf`, 2008. 24

[Nissim17] Kobbi Nissim, Thomas Steinke, Alexandra Wood, Mark Bun, Marco Gaboardi, David R OâBrien & Salil Vadhan. *Differential Privacy : A Primer for a Non-technical Audience*. 2017. 94, 101

[Odlyzko03] Andrew Odlyzko. *Privacy, economics, and price discrimination on the Internet*. In Proceedings of the 5th international conference on Electronic commerce, pages 355–366. ACM, 2003. 18

[oec18a] *Lignes directrices régissant la protection de la vie privée et les flux transfrontières de données à caractère personnel*. `http://www.oecd.org/`, 2018. Accessed : 2018-01-20. 15

[oec18b] *OECD Privacy Principles*. `http://oecdprivacy.org/`, 2018. Accessed : 2018-01-20. 15, 16

[Oh08] Seog-Chan Oh, Dongwon Lee & Soundar RT Kumara. *Effective web service composition in diverse and large-scale service networks*. IEEE Transactions on Services Computing, vol. 1, no. 1, pages 15–32, IEEE, 2008. 80

[ope10] *OpenDNS 2010 Report : Web Content Filtering and Phishing*. `https://athens.indymedia.org/media/old/opendns_report_2010.pdf`, 2010. Accessed : 2018-01-19. 42, 45

[Oulasvirta14] Antti Oulasvirta, Tiia Suomalainen, Juho Hamari, Airi Lampinen & Kristiina Karvonen. *Transparency of intentions decreases privacy concerns in ubiquitous surveillance*. Cyberpsychology, Behavior, and Social Networking, vol. 17, no. 10, pages 633–638, Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, 2014. 11

[Parsons15] June Jamrich Parsons. *Bundle : New Perspectives on Computer Concepts 2016, Comprehensive*. pages 273–274, Course Technology Press, 2015. 20

[Pfitzmann10] Andreas Pfitzmann & Marit Hansen. *A terminology for talking about privacy by data minimization : Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management*. 2010. 16

[phi18] *phishtank*. `https://www.phishtank.com/`, 2018. Accessed : 2018-01-19. 35, 37

[Pisinger95] David Pisinger. *A minimal algorithm for the multiple-choice knapsack problem*. European Journal of Operational Research, vol. 83, no. 2, pages 394–410, Elsevier, 1995. 69

[Posner81] Richard A Posner. *The economics of privacy*. The American economic review, vol. 71, no. 2, pages 405–409, JSTOR, 1981. 12

[Powers03] Calvin Powers & Matthias Schunter. *Enterprise privacy authorization language (EPAL 1.2)*. W3C Member Submission, vol. 10, 2003. 31

[PRIME05] Consortium PRIME*et al. Privacy and Identity Management for Europe (PRIME)*. Web site at www.prime-project.eu, 2005. 24

[Primepq08] Primepq. *diagramm : Basic decryption mix net*. `https://en.wikipedia.org/wiki/File:Decryption_mix_net.png`, 2008. vii, 28

[Quinlan14] J Ross Quinlan. C4. 5 : programs for machine learning. Elsevier, 2014. 109

[Quisquater89] Jean-Jacques Quisquater, Myriam Quisquater, Muriel Quisquater, Michaël Quisquater, Louis Guillou, Marie Annick Guillou, Gaïd Guillou, Anna Guillou, Gwenolé Guillou & Soazig Guillou. *How to explain zero-knowledge protocols to your children*. In Conference on the Theory and Application of Cryptology, pages 628–631. Springer, 1989. 26

[Rannenberg09] Kai Rannenberg, Denis Royer & André Deuker. The future of identity in the information society : Challenges and opportunities. Springer Science & Business Media, 2009. 24

[Recordon06] David Recordon & Drummond Reed. *OpenID 2.0 : a platform for user-centric identity management*. In Proceedings of the second ACM workshop on Digital identity management, pages 11–16. ACM, 2006. 23

[Reddy11] Venkata Prasad Reddy, V Radha & Manik Jindal. *Client Side protection from Phishing attack*. International Journal of Advanced Engineering Sciences and Technologies (IJAEST), vol. 3, no. 1, pages 39–45, 2011. 35, 37

[Reiss84] Steven P Reiss. *Practical data-swapping : The first steps*. ACM Transactions on Database Systems (TODS), vol. 9, no. 1, pages 20–37, ACM, 1984. 98

[Reiter98] Michael K Reiter & Aviel D Rubin. *Crowds : Anonymity for web transactions*. ACM transactions on information and system security (TISSEC), vol. 1, no. 1, pages 66–92, ACM, 1998. 27, 28

[Reiter03] Jerome P Reiter. *Inference for partially synthetic, public use microdata sets*. Survey Methodology, vol. 29, no. 2, pages 181–188, 2003. 102

[Reiter05] Jerome P Reiter. *Using CART to generate partially synthetic public use microdata*. Journal of Official Statistics, vol. 21, no. 3, page 441, Statistics Sweden (SCB), 2005. 102

[Rezgui02] Abdelmounaam Rezgui, Mourad Ouzzani, Athman Bouguettaya & Brahim Medjahed. *Preserving privacy in web services*. In Proceedings of the 4th international workshop on Web information and data management, pages 56–62. ACM, 2002. 52, 56

[Ritter13] Tom Ritter. *The Differences Between Onion Routing and Mix Networks*. `https://crypto.is/blog/mix_and_onion_networks`, 2013. Accessed : 2018-03-10. 28

[Rubin93] Donald B Rubin. *Discussion statistical disclosure limitation*. Journal of official Statistics, vol. 9, no. 2, page 461, Statistics Sweden (SCB), 1993. 101

[Rubner00] Yossi Rubner, Carlo Tomasi & Leonidas J Guibas. *The earth mover's distance as a metric for image retrieval*. International journal of computer vision, vol. 40, no. 2, pages 99–121, Springer, 2000. 101

[Run12] Cui Run, Hyoung Joong Kim, Dal-Ho Lee, Cheong Ghil Kim & Kuinam J Kim. *Protecting Privacy Using K-Anonymity with a Hybrid Search Scheme*. International Journal of Computer and Communication Engineering, vol. 1, no. 2, page 155, IACSIT Press, 2012. 100

[Salem10] Omran Salem, Alamgir Hossain & M Kamala. *Awareness program and ai based tool to reduce risk of phishing attacks*. In Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on, pages 1418–1423. IEEE, 2010. 36, 38

[Salton79] Gerard Salton. *Mathematics and information retrieval*. Journal of Documentation, vol. 35, no. 1, pages 1–29, MCB UP Ltd, 1979. 41

[Samarati98] Pierangela Samarati & Latanya Sweeney. *Generalizing data to provide anonymity when disclosing information*. In PODS, volume 98, page 188, 1998. 94, 99

[Samarati01] Pierangela Samarati. *Protecting respondents identities in microdata release*. IEEE transactions on Knowledge and Data

Engineering, vol. 13, no. 6, pages 1010–1027, IEEE, 2001. 94, 99

[Sarma08] Amardeo Sarma, Alfredo Matos, João Girão & Rui L Aguiar. *Virtual identity framework for telecom infrastructures*. Wireless Personal Communications, vol. 45, no. 4, pages 521–543, Springer, 2008. 23

[Schwaig06] Kathy Stewart Schwaig, Gerald C Kane & Veda C Storey. *Compliance to the fair information practices : How are the Fortune 500 handling online privacy disclosures ?* Information & management, vol. 43, no. 7, pages 805–820, Elsevier, 2006. 11

[Service17] Algérie Presse Service. *Adoption d'un projet de loi sur la protection des personnes physiques dans le traitement des données personnelles*. http://www.aps.dz/algerie/67623-adoption-d-un-projet-de-loi-sur-la-protection\-des-personnes-physiques-dans-le-traitement-\des-donnees-personnelles, 2017. Accessed : 2018-01-15. 14

[Sheng14] Quan Z Sheng, Xiaoqiang Qiao, Athanasios V Vasilakos, Claudia Szabo, Scott Bourne & Xiaofei Xu. *Web services composition : A decade's overview*. Information Sciences, vol. 280, pages 218–238, Elsevier, 2014. 123

[Simon04] Hank Simon. *SAML : The Secret to Centralized Identity Management*. Dec, 2004. 23

[Singhal01] Amit Singhal*et al. Modern information retrieval : A brief overview*. IEEE Data Eng. Bull., vol. 24, no. 4, pages 35–43, 2001. 41

[soc18] *What is Social Engineering ?* https://www.social-engineer.org/about/, 2018. Accessed : 2018-01-10. 36

[Solove02] Daniel J Solove. *Conceptualizing privacy*. Cal. L. Rev., vol. 90, page 1087, HeinOnline, 2002. 11, 12

[Solove06] Daniel J Solove. *A taxonomy of privacy*. U. Pa. L. Rev., vol. 154, page 477, HeinOnline, 2006. 17

[Squicciarini13] Anna Cinzia Squicciarini, Barbara Carminati & Sushama Karumanchi. *Privacy aware service selection of composite web services invited paper*. In 9th International Conference Conference on Collaborative Computing : Networking, Applications

and Worksharing (Collaboratecom), pages 260–268. IEEE, 2013. 52, 56

[Standard05] OASIS Standard. *extensible access control markup language (xacml) version 2.0*, 2005. 30, 31

[Stufflebeam04] William H Stufflebeam, Annie I Antón, Qingfeng He & Neha Jain. *Specifying privacy policies with P3P and EPAL : lessons learned*. In Proceedings of the 2004 ACM workshop on Privacy in the electronic society, pages 35–35. ACM, 2004. 31

[Sui17] Peipei Sui & Xianxian Li. *A privacy-preserving approach for multimodal transaction data integrated analysis*. Neurocomputing, vol. 253, pages 56–64, Elsevier, 2017. 22

[Sweeney02] Latanya Sweeney. *Achieving k-anonymity privacy protection using generalization and suppression*. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 05, pages 571–588, World Scientific, 2002. 94, 99

[Tbahriti14] Salah-Eddine Tbahriti, Chirine Ghedira, Brahim Medjahed & Michael Mrissa. *Privacy-Enhanced Web Service Composition*. IEEE Transactions on Services Computing, vol. 7, no. 2, pages 210–222, IEEE, 2014. 52, 54

[Team14] R Core Team. *R : A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2012*, 2014. 73

[Truta06] Traian Marius Truta & Bindu Vinay. *Privacy protection : p-sensitive k-anonymity property*. In Data Engineering Workshops, 2006. Proceedings. 22nd International Conference on, pages 94–94. IEEE, 2006. 100

[ukg17] *Information on identity theft*. `http://www.identitytheft.org.uk/`, 2017. Accessed : 2018-01-15. 18

[Vogt07] Philipp Vogt, Florian Nentwich, Nenad Jovanovic, Engin Kirda, Christopher Kruegel & Giovanni Vigna. *Cross Site Scripting Prevention with Dynamic Data Tainting and Static Analysis*. In NDSS, volume 2007, page 12, 2007. 20

[Wang08] Yue Wang, Rinky Agrawal & Baek-Young Choi. *Light weight anti-phishing with user whitelisting in a web browser*. In Region 5 Conference, 2008 IEEE, pages 1–4. IEEE, 2008. 35, 38

[Wang10] K Wang, R Chen, BC Fung & PS Yu. *Privacy-preserving data publishing : A survey on recent developments*. ACM Computing Surveys, 2010. 21, 22, 96, 99

[Warren90] Samuel D Warren & Louis D Brandeis. *The right to privacy*. Harvard law review, pages 193–220, JSTOR, 1890. 11

[Watts98] Duncan J Watts & Steven H Strogatz. *Collective dynamics of âsmall-worldânetworks*. nature, vol. 393, no. 6684, pages 440–442, Nature Publishing Group, 1998. 80

[Westin68] Alan F Westin. *Privacy and freedom*. Washington and Lee Law Review, vol. 25, no. 1, page 166, 1968. 11, 12

[Westin03] Alan F Westin. *Social and political dimensions of privacy*. Journal of social issues, vol. 59, no. 2, pages 431–453, Wiley Online Library, 2003. 11

[Whiting13] Jim Whiting. Identity theft. ReferencePoint Press, Inc., 2013. 17

[wom18] *2018 State of the Phish*. `https://www.wombatsecurity.com/state-of-the-phish`, 2018. Accessed : 2018-03-05. 21

[Wong06] Raymond Chi-Wing Wong, Jiuyong Li, Ada Wai-Chee Fu & Ke Wang. *(α, k)-anonymity : an enhanced k-anonymity model for privacy preserving data publishing*. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 754–759. ACM, 2006. 100

[Wu12] Quanwang Wu, Qingsheng Zhu & Peng Li. *A caching mechanism for QoS-aware service composition*. Journal of Web Engineering, vol. 11, no. 2, pages 119–130, Rinton Press, Incorporated, 2012. 52

[Xu06] Wei Xu, VN Venkatakrishnan, R Sekar & IV Ramakrishnan. *A framework for building privacy-conscious composite web services*. In International Conference on Web Services, ICWS'06., pages 655–662. IEEE, 2006. 52, 55

[Yan17] Ping Yan & Zheng Yan. *A survey on dynamic mobile malware detection*. Software Quality Journal, pages 1–29, Springer, 2017. 19

[Ye17] Yanfang Ye, Tao Li, Donald Adjeroh & S Sitharama Iyengar. *A survey on malware detection using data mining techniques*. ACM Computing Surveys (CSUR), vol. 50, no. 3, page 41, ACM, 2017. 19

[Yu07] Tao Yu, Yue Zhang & Kwei-Jay Lin. *Efficient algorithms for Web services selection with end-to-end QoS constraints*. ACM Transactions on the Web (TWEB), vol. 1, no. 1, page 6, ACM, 2007. 52

[Zhang05] Harry Zhang. *Exploring conditions for the optimality of naive Bayes*. International Journal of Pattern Recognition and Artificial Intelligence, vol. 19, no. 02, pages 183–198, World Scientific, 2005. 109

[Zhang07] Yue Zhang, Jason I Hong & Lorrie F Cranor. *Cantina : a content-based approach to detecting phishing web sites*. In Proceedings of the 16th international conference on World Wide Web, pages 639–648. ACM, 2007. 39, 43, 47

[Ziegeldorf14] Jan Henrik Ziegeldorf, Oscar Garcia Morchon & Klaus Wehrle. *Privacy in the Internet of Things : threats and challenges*. Security and Communication Networks, vol. 7, no. 12, pages 2728–2742, Wiley Online Library, 2014. 18