Pablo Rauzy

✉ pablo.rauzy@inria.fr
http://pablo.rauzy.name/

**Projets recherche et enseignement** – *Candidature MdC 27ème*

# Programme de recherche

## Approche formelle de la sécurité offensive et application à la protection de la vie privée

**Résumé en une page :**

La **cyber-sécurité** est aujourd'hui un enjeu majeur. La **sécurité offensive** consiste à attaquer le système que l'on cherche à protéger pour mieux comprendre ses faiblesses. C'est une méthode qui a fait ses preuves. Elle est largement utilisée dans l'industrie. En cryptologie, le principe même de la cryptanalyse est justement de casser les algorithmes et protocoles cryptographiques. Toutefois, l'approche offensive ne permet pas de prouver de façon rigoureuse la sécurité d'un système.

L'**approche formelle** consiste à prouver mathématiquement la sécurité d'un système, et vise aussi l'automatisation de la mise en œuvre de contre-mesures. Elle est requise pour atteindre les plus hauts niveaux de certification, par exemple pour les évaluations de niveaux d'assurance 5 à 7 des Critères Communs [Cri]. Cependant, les méthodes formelles posent des problèmes de passage à l'échelle.

Le but de mon projet de recherche est de **combiner ces deux approches, formelle et expérimentale**, pour tirer le meilleur parti de chacune.

La conception d'un **système d'information de confiance** demande non seulement de le protéger contre les agressions venant de l'extérieur, mais aussi de s'assurer que les informations qu'il contient ne pourront pas être utilisées contre la volonté de ses utilisateurs ou en dehors du cadre prévu par ses concepteurs. C'est pourquoi **le respect et la protection de la vie privée** est désormais un axe essentiel de la recherche en sécurité informatique. En particulier, il est devenu indispensable de protéger les **données personnelles** des individus et donc de comprendre et protéger les **systèmes** sur lesquels elles vivent et les **réseaux** sur lesquels elles transitent. Pour des raisons évidentes, les instituts de recherche publics et indépendants tels que les universités doivent jouer un rôle central dans ce domaine.

Je me propose de travailler sur les méthodes et outils qui permettent de concevoir des systèmes sécurisés et protecteurs de la vie privée par construction. Cela demande d'une part de formaliser ce qu'est le respect et la protection de la vie privée, et d'autre part de mettre au point une nouvelle approche formelle de la sécurité offensive.

Mon projet est organisé en deux axes. Le premier est le développement de méthodes formelles pour l'**analyse systématique de vulnérabilités**, et la preuve de contre-mesures. Il est décomposé en trois phases : (1) le développement de **modèles formels de vulnérabilités** ; (2) la mise au point d'**analyses hybrides** permettant d'exploiter à la fois la sémantique du code source et les techniques de rétro-ingénierie d'exécutables de logiciels malveillants ; (3) l'introduction de **scénarios d'attaques** dans les modèles de vulnérabilités.

Le second axe consiste à **appliquer à la protection de la vie privée** les résultats du premier axe. Il est décomposé en trois phases : (1) un travail fondamental de **formalisation de la vie privée** ; (2) un travail de **modélisation des flux de données personnelles dans les systèmes distribués** ; (3) enfin, l'application de l'**approche offensive de la sécurité** à la **protection de la vie privée**. La première phase implique notamment de modéliser le contrôle des utilisateurs sur leurs données personnelles, ainsi que les possibilités d'inférence d'informations par un attaquant. L'objectif est de créer une **mesure de respect et protection de la vie privée**. Le but de la deuxième phase est de **comprendre la collecte des données personnelles**, qui est aujourd'hui particulièrement opaque. Elle a lieu partout et tout le temps, sans que les individus en ait nécessairement conscience : réseaux mobiles, internet des objets, réseaux de capteurs, systèmes pairs-à-pairs, ville intelligente (e.g., *smart grid*), systèmes cyber-physiques (e.g., voitures connectées, domotique), etc. La troisième phase consiste justement à appliquer très concrètement dans ces systèmes et réseaux qui manipulent des données personnelles les méthodes et outils développés.

## Contexte

La sécurité de l'information, qui rassemble la protection de la vie privée des personnes et la sécurité des systèmes, est un domaine à plusieurs facettes. Il comporte notamment une grande part d'ingénierie, étant donnée la nécessité de protéger des systèmes d'information fortement complexes. En effet, les systèmes d'information du monde réel sont hétérogènes et interconnectés, et chaque partie repose sur plusieurs couches matérielles et logicielles. Un exemple typique de système d'information complexe et omniprésent est le téléphone : c'est un système fortement connecté aux réseaux Internet et GSM, sur lequel plusieurs applications ayant des droits différents sont exécutées parfois dans une machine virtuelle, parfois de manière isolée, tout en pouvant faire appel à des bibliothèques logicielles partagées et à des modules de sécurité matériels (pour des questions de performance). La nature même des systèmes de ce type les rend, à l'heure actuelle, hors de portée des méthodes formelles. Cela s'explique à la fois par leur taille, qui provoque des explosions combinatoires que l'on n'est pas capable de gérer en temps raisonnable, et par leur nature, qui les rend difficilement modélisables alors qu'une modélisation mathématique est une condition nécessaire à l'application de méthodes formelles.

Paradoxalement, c'est justement la complexité de ces systèmes qui rend l'utilisation de méthodes formelles indispensable. Bien qu'elle permette de découvrir de nouvelles vulnérabilités, l'application exclusive de techniques d'ingénierie a toujours montré ses limites en terme de sécurité comme en terme d'efficacité. Si les méthodes formelles sont plus coûteuses, elles permettent néanmoins de prouver les propriétés de sécurité désirées du système étudié, et ainsi augmenter la confiance que l'on peut avoir dans le système prouvé. Du même coup, ces méthodes autorisent bien souvent l'optimisation du système. En effet, sans possibilité de vérifications et de tests automatiques des propriétés de sécurité, il n'est pas raisonnable d'optimiser un système informatique critique : chaque optimisation introduit potentiellement des vulnérabilités. Pourtant, la nécessité de l'optimisation, hier dans les systèmes embarqués contraints en ressources, et demain pour l'« internet des objets », est telle qu'elle prime souvent sur la sécurité dans le processus de développement.

Dans le domaine de la sécurité de l'information, il reste encore énormément de zones inexplorées par l'approche formelle qui pourraient pourtant en bénéficier grandement. Par exemple, les méthodes formelles n'en sont qu'à leurs balbutiements sur tout ce qui peut être apparenté à des canaux auxiliaires, c'est-à-dire ce qui n'est pas représenté dans les modèles classiques de sécurité et qui ne semble à première vue que difficilement modélisable. C'est bien évidemment vrai concernant la sécurité physique des implémentations cryptographiques. C'est également le cas dans les applications réseaux, où par exemple la taille des paquets transmis contient bien plus d'information qu'on ne pourrait l'imaginer [SSH+14]. C'est encore vrai des problématiques liées au respect et à la protection de la vie privée qui, du point de vue des canaux auxiliaires, peuvent correspondre à des fuites de données corrélées [BHJ+13], ou encore être simplement des fuites directes d'informations sensibles qui ne sont pourtant pas considérées sensibles au moment du développement des systèmes. Par exemple, un téléphone émet régulièrement la liste des réseaux WiFi qu'il connaît, qui, en plus d'être souvent unique, révèle très directement une liste de lieux que le téléphone a visités [SMOS12]. De même, les méthodes formelles ne sont encore que peu (voire pas) utilisées dans le domaine de la sécurité offensive, qui consiste en la recherche de failles dans les systèmes d'information afin de pouvoir mieux comprendre l'origine des vulnérabilités et de pouvoir les en protéger. Pourtant, comme l'expliquait Jean-Louis Lanet [Lan15], l'approche offensive de la sécurité bénéficierait de la rigueur et de la systématisation des techniques qu'apporterait une approche formelle.

Assurer la sécurité d'un système est une tâche très difficile, et il ne peut y avoir de sécurité réelle que si celle-ci est considérée de manière globale. Malheureusement, nous sommes encore très loin de pouvoir étudier formellement la sécurité d'un système globalement, si tant est qu'il soit possible d'y parvenir complètement. Il faudrait pour cela des modèles précis du système allant de la sécurité physique de son implémentation jusqu'à la protection de la vie privée de ses utilisateurs, en passant par ses couches système, réseau, protocole, et application. Il faudrait aussi qu'à chaque niveau, ces modèles autorisent la prise en compte des différents types d'attaques, y compris par exemple des attaques actives [1]. À mon sens, la mise au point de telles abstractions doit être l'objectif à long terme de la communauté scientifique de la sécurité de l'information. Dans cette optique, il me semble nécessaire de commencer par faire des efforts de modélisation à chaque niveau, pour la mise en œuvre de méthodes formelles

1. Par opposition aux attaques passives, où l'attaquant se contente d'« écouter » des fuites d'informations.

permettant l'étude des différentes briques qui composent le système. Ensuite, il faudra faire le lien entre les modèles qui vont être créés, et s'assurer de leur composabilité les uns avec les autres. À terme, le but est de proposer une chaîne d'outils permettant la conception de systèmes qui soient par construction à la fois sécurisés, et respectueux et protecteurs de la vie privée.

## Axes de recherche

Mon projet de recherche s'articule autour de deux axes plus ou moins orthogonaux, tentant de répondre à une partie des problématiques de sécurité exposées dans le contexte : l'analyse de vulnérabilité et la protection de la vie privée. Plus précisément, je souhaite m'attaquer à ces sujets au niveau logiciel des plateformes mobiles (téléphones, tablettes, etc.) et plus largement des objets connectés, y compris des systèmes cyber-physiques, ces systèmes critiques de plus en plus composés eux-mêmes de multiples plateformes complexes organisées en réseau. Il existe des points communs entre toutes ces plateformes qui sont à la fois les raisons de leurs faiblesses en sécurité, et ce qui permet le développement de modèles réutilisables d'une plateforme à l'autre. Ces points communs sont entre autres : la connectivité accrue au réseau Internet agrandissant la surface de vulnérabilités potentielles, l'augmentation de la masse de données personnelles manipulées faisant de fait de ces plateformes des systèmes d'information critiques, et la complexification de ces systèmes et réseaux qui aujourd'hui font usage de matériels (processeurs multi-cœurs, composants spécialisés) et de systèmes d'exploitation complexes (typiquement Android) dans des réseaux très dynamiques.

L'**analyse de vulnérabilités** pour la sécurisation de systèmes logiciels est la thématique l'axe principal. L'objectif de cet axe est le développement d'approches formelles pour la découverte systématique de vulnérabilités, et la preuve de contre-mesures. Nous distinguons trois phases. La première phase de cet axe consiste en la mise au point de *modèles formels de vulnérabilités*. Ces modèles serviront de socle à la vérification automatisée par *analyses statiques et dynamiques* de la présence des types de vulnérabilités modélisées. La deuxième phase consiste à faire le *lien entre le comportement de l'exécutable et de son code source*, c'est-à-dire entre les traces d'exécution du logiciel et sa sémantique de haut niveau (y compris en utilisant des méthodes de *rétroconception en boîte blanche*), afin de concevoir des méthodes d'*analyse hybride* faisant usage en même temps de la sémantique du code source et des techniques d'analyse d'exécutables. Enfin, la troisième phase consiste en l'introduction de *scénarios d'attaques* dans les modèles, pour étudier leur évolution en présence par exemple d'attaques pouvant modifier le système en cours d'exécution [2], en plus de pouvoir comme auparavant utiliser les entrées du système. La modélisation formelle de vulnérabilités permettra l'identification de possibles *contre-mesures*, voire leur insertion automatisée. L'efficacité d'une contre-mesure en terme de sécurité face à un type d'attaque rigoureusement défini pourra être prouvée par couverture complète et systématique de ces attaques grâce à des analyses de vulnérabilités avant et après application de la contre-mesure.

L'**application à la protection de la vie privée** des résultats du premier axe est l'objectif du second axe. L'idée est de mettre en œuvre les méthodes et outils de détection de vulnérabilités pour exposer les fuites d'informations personnelles, et ainsi sensibiliser les utilisateurs. La première phase de cet axe correspond à un travail fondamental de *formalisation de la vie privée*, notamment de modélisation du contrôle des utilisateurs sur leurs données personnelles, ainsi que des possibilités d'*inférences d'informations* par l'attaquant, dans le but de pouvoir créer une *mesure* de respect et protection de la vie privée. La deuxième phase consiste en l'étude des flux de données personnelles dans les réseaux dynamiques, qu'il s'agisse de réseaux sociaux ou de réseaux concrets dans le cas des objets connectés (internet des objets, plateformes mobiles, systèmes cyber-physiques) ou des réseaux type pairs-à-pairs de technologies visant à renforcer le respect de la vie privée [3] comme les logiciels Tor [4] et I2P [4], par exemple. La troisième phase consiste à appliquer l'*approche offensive de la sécurité à la protection de la vie privée*, notamment en utilisant les méthodes du premier axe sur des systèmes manipulant des données personnelles que l'on vient de citer. Le but de cet axe est aussi de rendre la sécurité, en particulier celle liée aux problématiques de vie privée, plus accessible et compréhensible de tous afin de permettre aux utilisateurs de faire des choix mieux informés dans leurs utilisations des technologies de l'information.

2. Par exemple, dans les phases 1 et 2 on veut détecter la possibilité d'une attaque par débordement de tampon, puis dans la phase 3 on étudie ses possibles exploitations (e.g., modification d'une variable, insertion de code exécutable).
3. En anglais, *privacy enhancing technologies*, ou PETs.
4. `https://www.torproject.org/`, `https://geti2p.net/`.

## Axe 1 — Analyses de vulnérabilités

Le recherche de vulnérabilités logicielles est une pratique de sécurité offensive. Une majeure partie de la littérature se concentre sur l'étude de maliciels [5], en cherchant à comprendre quelles failles ils exploitent. Cette méthode de découverte de vulnérabilités est efficace, cependant, il serait plus intéressant de découvrir les vulnérabilités avant qu'elles ne soient exploitées. C'est l'objectif de cet axe de recherche.

### Phase 1 – Modèles formels de vulnérabilités

Aujourd'hui, les techniques d'analyse statique de code source permettent d'éviter de nombreux bugs fonctionnels, mais ne permettent pas encore de s'assurer de l'absence de vulnérabilité dans le logiciel compilé. Des tentatives d'application des méthodes d'analyse statique à la recherche de vulnérabilités existent. Par exemple, l'interprétation abstraite peut être utilisée pour découvrir les possibilités de débordement de tampon [DRS03]. Mais ces méthodes, en plus d'être modérément efficaces (problème de faux positifs), ne passent pas à l'échelle.

Il existe déjà des techniques semi-automatisées permettant l'étude et la détection de maliciels, comme en témoignent les travaux de Christodorescu et al. [CJK07]. En revanche, s'il existe plusieurs travaux, comme les thèses de Ozment [Ozm07] et Krsul [Krs98], proposant des classifications *a posteriori* de vulnérabilités, leur modélisation formelle *a priori* est encore inexistante. Pour l'instant, la découverte de vulnérabilités est principalement faite à la main par des experts. Quand elle est semi-automatisée, elle relève de techniques de force-brute ou de tests à données aléatoires [6] (voir à ce sujet le livre de Sutton, Green et Amini [SGA07]) guidées par la connaissance et l'habitude des experts.

Le but de cette phase est de rendre efficace la vérification automatique de la présence d'un type de vulnérabilité modélisé. Pour cela, il faut d'abord combler le manque de modèles formels de vulnérabilités. Ceux-ci pourraient être dérivés des travaux cités, notamment en croisant les classifications de vulnérabilités et les comportements typiques des programmes dans lesquels elles apparaissent.

Les approches les plus prometteuses en matière d'analyse d'exécutables semblent être des analyses dynamiques : le test concolique [7] introduit par Godefroid et al. dans [GKS05], ou encore l'analyse dynamique teintée [8] et l'exécution symbolique « en avant » [8] (expliquées toutes les deux en détails par Schwartz et al. dans [SAB10]). L'idée du test concolique est de lancer en parallèle une exécution symbolique et une exécution concrète sur des entrées particulières, pouvant provenir de contraintes générées par l'exécution symbolique, dans le but de se rapprocher d'une couverture maximale du code. L'analyse dynamique teintée permet de suivre le flot de certaines informations choisies (teintées) depuis les entrées du programme. L'exécution symbolique « en avant » permet, quant à elle, d'étudier le comportement d'un programme sur beaucoup d'entrées à la fois en représentant son exécution par des formules logiques.

Ainsi, grâce à ces modèles, les techniques existantes et efficaces de détection de maliciels pourraient être mises à profit pour détecter des vulnérabilités. Je pense entre autres aux travaux de Song et Touilli [ST12, ST13] qui modélisent les comportements de maliciels en les formalisant comme des automates à pile puis les détectent à l'aide de vérification de modèle [9]. Il existe également des méthodes de détection de vulnérabilités spécifiques, notamment dans le cadre des injections SQL dans les applications web, comme les travaux de Alata et al. [LBA14, AAKN14], qui pourraient être adaptés à d'autres types de vulnérabilités une fois celles-ci proprement modélisées.

En revanche, une approche purement formelle de la recherche de vulnérabilités serait intrinsèquement limitée. D'un côté, on ne peut modéliser formellement que ce qu'on connaît, et de l'autre, les vulnérabilités d'un système se trouvent par définition le plus souvent en dehors de ses spécifications. L'approche offensive de la sécurité permet de combler cette lacune en attaquant les systèmes avec des méthodes d'ingénieries et en boîte noire. Cela permet, une fois qu'un nouveau type de vulnérabilités est découvert, d'envisager sa modélisation formelle.

5. Logiciels malveillants. En anglais, *malware*.
6. En anglais, *fuzzing*.
7. Mot-valise composé à partir des mots anglais *concrete* et *symbolic*.
8. En anglais, *dynamic tainted execution* et *forward symbolic execution*.
9. En anglais, *model checking*.

**Phase 2 – Analyse hybride**

Une fois les vulnérabilités modélisées, il s'agit d'utiliser le plus efficacement possible ces modèles pour la détection de failles. L'analyse statique de codes sources, l'analyse dynamique de code machine, et l'exécution symbolique sont des approches complémentaires qui ont chacune fait leurs preuves. D'un côté l'analyse dynamique d'exécutables et leur exécution symbolique (déjà rassemblées dans l'analyse concolique) permettent d'identifier concrètement les opérations critiques du programme (accès mémoire, appels système, etc.). D'un autre côté, la sémantique du code source permet de raisonner plus efficacement et à plus haut niveau. Le but de cette phase est de développer des méthodes et des outils d'analyse hybride tirant parti à la fois du code source haut niveau et de l'exécution du code machine.

Par exemple, l'analyse hybride exécutable/source permettrait de faire de l'*analyse teintée à rebours*, en se basant sur l'idée de l'exécution dynamique teintée [NS05], mais en utilisant la sémantique du code source haut niveau plutôt que l'exécution symbolique du code machine pour pouvoir remonter aux entrées permettant d'exploiter une vulnérabilité potentielle une fois celle-ci détectée.

> Si l'on dispose de modèles de comportements d'exécution typiques permettant d'opérer une injection de code SQL, et que l'on détecte un tel comportement lors de l'analyse dynamique de l'exécutable, une analyse statique du code source correspondant permettrait d'augmenter la précision de la description de la vulnérabilité potentielle. On pourra alors calculer la grammaire de ce qu'il serait possible d'injecter dans la requête.
>
> Cela peut servir à *détecter des faux positifs* : grâce au filtrage des entrées, la grammaire de l'injection ne permet pas d'injecter du code qui soit considéré problématique. Ainsi, si le typage du code source révèle que l'entrée est sujette à un hachage cryptographique avant d'être insérée dans la requête SQL, son format très strict ne posera pas de problème. Pourtant le condensé aurait été teinté en suivant simplement le flot d'information lors de l'exécution du code machine, sans l'information sémantique apportée par le typage du code haut niveau.
>
> Mais cela peut aussi *confirmer et mieux caractériser la vulnérabilité* : en comprenant à la fois le sens de la requête SQL et la grammaire des données injectables, on peut se rendre compte que seul l'ajout de tuples incorrects dans la base est possible mais pas la suppression de données, par exemple.

La *rétroconception en boîte blanche*, c'est-à-dire la mise en correspondance du comportement d'un exécutable avec son code source [10] est une approche possible pour lier les modèles formels de comportement d'exécutables à la sémantique du code source. Ce serait en même temps une première étape en direction de l'objectif à long terme de lier les modèles formels à tous les niveaux d'un système pour pouvoir étudier sa sécurité globalement.

Une autre piste explorable connexe à cette phase de recherche est ainsi la désobfuscation de code. L'obfuscation de code consiste à rendre très difficile l'accès au code source d'un exécutable, pour empêcher la compréhension de son fonctionnement (par exemple rendre compliquée l'analyse d'un maliciel). Le travail que je souhaite réaliser sur l'analyse hybride exécutable/source permettra de faire correspondre des motifs de comportement de l'exécutable et de son code source haut niveau, permettant ainsi d'aider à la désobfuscation de code. En retour, la rétroconception de « composant pris sur étagère » [11] permettrait de les assujettir à l'analyse hybride pour la détection de vulnérabilités.

Les résultats de cette phase de recherche permettraient de développer des méthodes et outils repoussant les capacités de l'état de l'art en matière d'analyse de vulnérabilités, représenté par des plateformes comme BAP [12] et BitBlaze [13]. De plus, le lien entre la sémantique du code source et la détection de vulnérabilités sera utile dans mon second axe de recherche : il aidera à faire le lien entre les modèles, nécessairement de très haut niveau, de respect et protection de la vie privée et les modèles de vulnérabilités.

10. Dont on dispose, par opposition à la rétroconception en boîte noire où il s'agit de retrouver le code source inconnu.
11. En anglais, *commercial off-the-shelf systems*, ou COTS.
12. Binary Analysis Platform, `http://bap.ece.cmu.edu/`.
13. BitBlaze binary analysis project, `http://bitblaze.cs.berkeley.edu/`.

**Phase 3 – Modélisation de scénarios d'attaques**

Dans cette phase de recherche, il s'agit d'aller plus loin dans la compréhension des vulnérabilités :

1. Que se passe-t-il si l'attaquant peut modifier des valeurs à d'autres endroits que les entrées du programme (injections de fautes) ? Même question pour la modification du code.
2. Quels sont les effets possibles de l'exploitation d'une vulnérabilité particulière ?
3. Que se passe-t-il si l'attaquant a déjà exploité une première vulnérabilité ?

Dans le cadre des attaques par injection de fautes sur des implémentations d'algorithmes de cryptographie asymétrique, j'ai montré pendant ma thèse [RG14a, RG14b] qu'il était possible de modéliser formellement ces attaques et ainsi de mettre en œuvre la découverte de vulnérabilités de manière systématique par une analyse statique des algorithmes. La première question peut être vue comme un élargissement de ces travaux à des implémentations autres que cryptographiques. Une collaboration avec le laboratoire Vérimag dans le cadre du projet Sertif [14] est déjà envisagée.

La deuxième question concerne l'utilisation effective des vulnérabilités d'un système. Il s'agit d'étudier comment et pour quels résultats une vulnérabilité peut être utilisée par un attaquant. Les méthodes développées dans la phase 2 de cet axe de recherche aideront à répondre à ce type de question, comme le montre l'exemple avec les injections SQL donné dans l'encadré gris de la page précédente.

La troisième question introduit la notion d'attaques complexes, c'est-à-dire exploitant plus d'une vulnérabilité, sachant que de nouvelles vulnérabilités peuvent apparaître à la suite de l'exploitation d'une première.

> Un exemple extrême est l'utilisation d'un *shellcode* : dans un premier temps, un débordement de tampon est exploité pour donner l'accès à un *shell* (i.e., une couche logicielle qui fournit l'interface utilisateur d'un système) pouvant permettre le contrôle total du système par l'attaquant.

Heureusement, toutes les vulnérabilités ne donnent pas un contrôle total à l'attaquant. Malheureusement, un attaquant déterminé va en pratique souvent faire plus qu'exploiter une unique vulnérabilité pour arriver à ses fins. L'étude d'attaques complexes est donc une nécessité, en particulier sur les plateformes mobiles et les objets connectés qui ont de grandes surfaces de vulnérabilité potentielle (plusieurs vecteurs de communications — WiFi, réseaux mobiles, NFC — des couches logicielles rarement voire jamais mises à jour, etc.) tout en contenant des données de valeur attirant les attaques (mots de passe de différents systèmes distants, coordonnées bancaires, données personnelles, etc.).

Dans cette optique, il est nécessaire non seulement de détecter les vulnérabilités, mais aussi de bien les comprendre pour savoir ce que leur exploitation implique. Comment alors faire évoluer les modèles de vulnérabilités pour prendre ces attaques en considération ? Le travail effectué dans la phase 2 de cet axe de recherche participera à cette compréhension. Il sera également possible de tirer parti des travaux de Alata et al. [AKNA13] sur la génération automatique de scénarios d'attaques contre les applications web.

## Axe 2 — Application à la protection de la vie privée

Le respect et la protection de la vie privée (abrégé « privacy » dans la suite) est un des aspects de plus haut niveau (en terme d'abstraction) de la sécurité de l'information. C'est un sujet difficile car il touche à l'humain : on ne cherche plus seulement à protéger des données brutes (comme en cryptologie) ou même de l'information structurée (comme en sécurité de l'information), mais aussi le sens de l'information, qui parfois ne peut se révéler que par corrélations et inférences à partir de données hétérogènes, et qui est de toutes manières souvent hors de portée des machines (à l'heure actuelle du moins).

Par essence, la privacy est un sujet qui rapproche l'informatique d'autres disciplines comme le droit ou la sociologie. Pour autant, le sujet ne s'éloigne pas de l'informatique, en particulier du domaine de la sécurité, comme le montre l'appel à la soumission d'articles du 29ème Symposium sur les fondements de la sécurité informatique [15] qui aura lieu en 2016 : une session spéciale sur la privacy est organisée, témoignant d'un intérêt certain de la communauté informatique internationale pour ce sujet.

14. Projet ANR-14-ASTR-0003-01, `http://sertif-projet.forge.imag.fr/fr/`.
15. 29th IEEE Computer Security Foundations Symposium, `http://csf2016.tecnico.ulisboa.pt/cfp.html`.

**Phase 1 – Formalisation de la privacy**

L'étude scientifique de la privacy en est encore à ses balbutiements. La littérature se concentre essentiellement autour de l'adaptation de solutions techniques, notamment en créant des variantes de contrôle d'accès. Typiquement, les travaux de Byun et Li [BL08] sur le contrôle d'accès en fonction des objectifs [16], ou encore ceux de Park et Sandhu [PS10] sur le contrôle d'usage. Une autre branche de la littérature se concentre sur la définition de langages de spécification de politiques de vie privée. Par exemple, les travaux de Becker et al. [BMB10] ou de Pearson et Mont [PM11]. Cependant, il n'existe à ma connaissance pas encore de cadre formel permettant d'exprimer les propriétés attendues et les exigences en terme de privacy. Or, je crois qu'il est important pour pouvoir raisonner correctement sur un sujet de commencer par le définir rigoureusement. C'est en tout cas une nécessité si l'on veut pouvoir automatiser l'analyse de la privacy dans un système d'information, ou même aller encore plus loin et prouver des propriétés du système liées à la privacy. La complexité du sujet appelle naturellement à l'utilisation de méthodes formelles, ce qui implique la nécessité de faire un effort de modélisation qui servira de socle au développement de ces méthodes.

Mon projet actuel en post-doctorat est un premier pas vers la modélisation de la privacy. Je cherche à comprendre ce que signifie concrètement le fait d'avoir le contrôle sur ses données personnelles, pour définir un *langage formel de description de ce contrôle* dans un système d'information. Le but à plus long terme est que ces modèles formels de systèmes permettent de tracer l'origine des potentielles pertes de contrôle pour les corriger, voire de certifier l'absence de tels soucis. Il s'agit dans cette phase de poursuivre puis de dépasser ce travail.

En effet, il est aussi nécessaire de modéliser d'autres aspects de la privacy que le contrôle sur les données personnelles. Par exemple, les possibilités d'*inférences de nouvelles informations à partir d'un jeu de données personnelles* sont un enjeu majeur. Des travaux ont déjà été entrepris dans ce sens-là, notamment par Huguenin et d'autres dans une série d'articles récents [VHBH14, OHSH14, VHBH13] qui se concentrent sur l'inférence de données personnelles de localisation.

Dans le but de pouvoir évaluer les risques liés à l'inférence de données personnelles en s'abstrayant au maximum des systèmes concrets dans lesquels vivent ces données, il reste cependant à généraliser ce genre d'approche à tous types de données personnelles, sans lier l'inférence à une technologie précise. Je crois qu'il faut pour cela bâtir entre autres sur les travaux réalisés autour du web sémantique, en particulier autour des grosses bases de connaissances semi-structurées telles que DBPedia [BLK+09], Wikidata [VK14], ou YAGO [SKW07]. Une *ontologie spécialisée* modélisant les types de données personnelles et leurs liens sémantiques permettrait de raisonner pour déterminer quelles données personnelles peuvent être reconstituées à partir des données dont on dispose.

La mise au point d'une *mesure* du niveau de privacy est aussi un enjeu majeur. C'est cette mesure qui permettra l'évaluation des risques ainsi que celle des solutions proposées. Des mesures ont déjà été proposées pour répondre à des problèmes spécifiques comme l'anonymisation de microdonnées [17], par exemple par Li et al. [LLV10]. Mais ce travail est très spécialisé et n'est pas généralisable à d'autres problèmes de privacy. À ce jour, aucune mesure de la privacy dans un système d'information n'a été proposée. Une telle mesure devra prendre en compte au moins trois dimensions orthogonales : (1) le contrôle des utilisateurs sur leurs données personnelles (en utilisant le modèle de contrôle développé dans cette même phase), (2) les possibilités d'inférence à partir des données présentes dans le système (par exemple en mettant à profit des outils venant de la théorie de l'information et une ontologie telle que celle décrite au paragraphe précédent), et (3) la sécurité du système (en utilisant les modèles de l'axe 1, des preuves de propriétés de sécurité, etc.). Il serait intéressant de regarder aussi du côté des méthodes d'apprentissage statistique pour aider à la conception d'une telle mesure.

**Phase 2 – Modélisation des flux de données dans les systèmes distribués**

Dans la phase précédente il s'agissait principalement de modéliser la privacy en fonction du contrôle des individus sur leurs données personnelles. Effectivement les données personnelles sont au centre des problématiques liées à la privacy. Malheureusement, on est rarement conscient de l'existence même de

16. En anglais, *purpose-based access control*.

17. En anglais, *microdata*, dans le cadre de réponses à un sondage, désigne une information au niveau d'un individu (par opposition aux résultats du sondage qui sont publiés de manière agrégée).

ces données. Ce manque d'information annule d'emblée tout effort de contrôle des données personnelles par les individus qu'elles concernent : il est donc indispensable de comprendre quand et où des données personnelles sont créées et collectées, et d'être capable de suivre leurs déplacements à partir de là.

Des travaux en rapport avec le suivis de données personnelles ont déjà été menés par différentes équipes [FG96, Ber03, SL00, LGH89]. En revanche, les solutions proposées ne sont plus adaptées au monde actuel, dans lequel des données personnelles sur un individu sont créées partout et en permanence, sans que celui-ci n'ai à faire quoi que ce soit pour, et donc sans qu'il en ait conscience. Il y a donc un travail conséquent de compréhension des flux de données personnelles dans les réseaux sur lesquelles elles transitent aujourd'hui : Internet des objets, *smart grid*, systèmes pairs-à-pairs, etc.

De plus, une bonne compréhension des déplacements de données personnelles dans les réseaux qu'elles empruntent permettrait d'identifier des sources de vulnérabilités (ici au sens de fuites de données personnelles) potentielles et donc de de discerner les origines de certaines pertes de contrôle des individus sur leurs données.

Le but ultime de ces deux premières phases du second axe est de faire fonctionner conjointement les modèles de vulnérabilités de l'axe 1 et les modèles de privacy, afin de pouvoir étudier les effets sur la privacy de la potentielle exploitation de failles logicielles dans les systèmes et réseaux. Le travaux fondamentaux sur la privacy proposé dans des deux phases serviront de socle pour passer à la troisième phase de cet axe de recherche.

**Phase 3 – Approche offensive de la privacy**

Cette phase est presque entièrement expérimental. Il s'agit de la mise en œuvre des méthodes et techniques développées dans l'axe 1 et les premières phases de l'axe 2. D'un côté, cela permet de les mettre à l'épreuve pour confirmer leur utilité et leur efficacité. Typiquement, vérifier si l'exploitation d'une vulnérabilité correspond en pratique à ce que prédisent les mesures théoriques de privacy sur les modèles. D'un autre côté, la mise en œuvre de démonstrations compréhensibles du grand public permettra sa sensibilisation aux problématiques de sécurité, en particulier celles liées à la privacy. En même temps, cette approche donne une ouverture et une visibilité souhaitables au laboratoire, à l'Université Paris 8, et à la recherche en sécurité de manière plus générale.

Les systèmes cibles sont les mêmes que dans le premier axe, à savoir principalement les plateformes mobiles et objets connectés. En plus de cela il serait intéressant de d'étudier aussi aux technologies visant à renforcer la privacy (comme Tor, ou I2P), ainsi qu'à des systèmes critiques comme les machines de vote électronique. Par ailleurs, les objets connectés en rapport avec l'informatique médicale (sur lesquels travaillent une partie des membres du LIASD) se prêtent parfaitement à l'exercice car ils comprennent aussi bien un aspect sécurité qu'un aspect privacy. Les vulnérabilités détectées dans ces systèmes seront bien sûr communiquées aux fabriquants selon le principe de la « divulgation responsable », afin de leur donner l'occasion de corriger les failles avant leurs publications.

De par sa nature, cette phase sera principalement composée de projets à court terme (moins d'un an pour certains) et à fort impact sociétal, tout en alliant la rigueur scientifique des méthodes formelles à leur application concrète. D'une part, cela permettra d'attirer de bons étudiants motivés par les problématiques de la recherche et ses applications dès leur stage de master. D'autre part, ces projets ouvrent la voie à des partenariats industriels avec des entreprises, notamment avec les nombreuses jeunes pousses technologiques autour des objets connectés et des plateformes mobiles.

Les travaux à plus long terme ou avec de multiples échéances de cette phase de recherche pourraient aussi être montés en projets communs avec la CNIL, sur le modèle du projet Mobilitics [18]. De même, ces travaux pourraient aussi se faire dans le cadre de projets en collaboration avec des industriels.

18. `http://planete.inrialpes.fr/~achara/mobilitics/`

## Calendrier prévisionnel

### Objectifs à court terme (1 à 2 ans)

— Assemblage d'un catalogue de modèles formels de types de vulnérabilités connues.
— Mise au point d'analyses hybrides (code source et binaire) de détection de ces vulnérabilités.
— Formalisation d'un modèle de contrôle des données personnelles et mise au point d'une mesure de ce contrôle.

### Objectifs à moyen terme (3 à 5 ans)

— Modélisation des flux de données personnelles et identification de potentielles vulnérabilités liées.
— Prise en compte de scénarios d'attaques dans les modèles de recherche de vulnérabilités.
— Prise en compte des risques de perte de contrôle de données personnelles liés aux vulnérabilités modélisées.
— Publication d'un outil d'analyse hybride et son utilisation pour l'analyse d'un système réel manipulant des données personnelles.

### Objectifs à long terme (5+ ans)

— Unification des modèles de vulnérabilités logicielles et de perte de contrôle sur les données personnelles pour créer un lien formel entre sécurité et protection de la vie privée, permettant à terme la mise en place de principes de développement de systèmes d'information *sécurisés par construction* et *respectueux et protecteurs de la vie privée par construction.*

## Références


[AAKN14] Rim Akrout, Éric Alata, Mohamed Kaâniche, and Vincent Nicomette. An automated black box approach for web vulnerability identification and attack scenario generation. *Journal of the Brazilian Computer Society*, 2014.

[AKNA13] Éric Alata, Mohamed Kaâniche, Vincent Nicomette, and Rim Akrout. An Automated Approach to Generate Web Applications Attack Scenarios, 2013.

[Ber03] W. Berson. System for finding, identifying, tracking, and correcting personal information in diverse databases, 2003. US Patent 6,532,459.

[BHJ+13] Igor Bilogrevic, Kévin Huguenin, Murtuza Jadliwala, Florent Lopez, Jean-Pierre Hubaux, Philip Ginzboorg, and Valtteri Niemi. Inferring social ties in academic networks using short-range wireless communications. In *12th annual ACM Workshop on Privacy in the Electronic Society*, pages 179–188. ACM, 2013.

[BL08] Ji-Won Byun and Ninghui Li. Purpose Based Access Control for Privacy Protection in Relational Database Systems. *The VLDB Journal*, pages 603–619, 2008.

[BLK+09] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia - a crystallization point for the Web of Data. *J. Web Semantics*, page 154–165, 2009.

[BMB10] Moritz Y. Becker, Alexander Malkis, and Laurent Bussard. S4P : A Generic Language for Specifying Privacy Preferences and Policies, 2010.

[CJK07] Mihai Christodorescu, Somesh Jha, and Christopher Kruegel. Mining Specifications of Malicious Behavior. In *Proceedings of the the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on The Foundations of Software Engineering*, pages 5–14. ACM, 2007.

[Cri] Common Criteria. Evaluation Assurance Level. Wikipédia `https://fr.wikipedia.org/wiki/Evaluation_Assurance_Level`.

[DRS03] Nurit Dor, Michael Rodeh, and Mooly Sagiv. CSSV : Towards a Realistic Tool for Statically Detecting All Buffer Overflows in C. In *Proceedings of the ACM SIGPLAN 2003 Conference on Programming Language Design and Implementation*, pages 155–167. ACM, 2003.

[FG96] Eric Freeman and David Gelernter. Lifestreams : A Storage Model for Personal Data. *SIGMOD Rec.*, 25(1), 1996.

[GKS05] Patrice Godefroid, Nils Klarlund, and Koushik Sen. DART : Directed Automated Random Testing. In *Proceedings of the 2005 ACM SIGPLAN Conference on Programming Language Design and Implementation*, pages 213–223. ACM, 2005.

[Krs98] Ivan Victor Krsul. Software Vulnerability Analysis, 1998. PhD thesis, Purdue University.

[Lan15] Lanet, Jean-Louis. Black hats can also benefits from formal methods, 2015. PROOFS 2015, invited talk. `http://proofs-workshop.org/slides/PROOFS2015_JeanLouis_LANET.pdf`.

[LBA14] Didier Le Botlan and Éric Alata. Formalisation et génération d'injections, 2014.

[LGH89] A.R. Lessin, F.M. Gruppuso, and S.A. Harrison. Intelligent portable interactive personal data system, 1989. US Patent 4,868,376.

[LLV10] Ninghui Li, Tiancheng Li, and S. Venkatasubramanian. Closeness : A New Privacy Measure for Data Publishing. *Knowledge and Data Engineering, IEEE Transactions on*, pages 1041–4347, 2010.

[NS05] James Newsome and Dawn Song. Dynamic Taint Analysis : Automatic Detection, Analysis, and Signature Generation of Exploit Attacks on Commodity Software. In *Proceedings of the Network and Distributed Systems Security Symposium*, 2005.

[OHSH14] Alexandra-Mihaela Olteanu, Kévin Huguenin, Reza Shokri, and Jean-Pierre Hubaux. Quantifying the Effect of Co-Location Information on Location Privacy. In *14th Privacy Enhancing Technologies Symposium (PETS)*, pages 2445–2457. IEEE, 2014.

[Ozm07] Andy Ozment. Vulnerability Discovery & Software Security, 2007. PhD thesis, University of Cambridge.

[PM11] Siani Pearson and Marco Casassa Mont. Sticky Policies : An Approach for Managing Privacy Across Multiple Parties. *Computer*, pages 60–68, 2011.

[PS10] Jaehong Park and Ravi Sandhu. A Position Paper : A Usage Control (UCON) Model for Social Networks Privacy, 2010.

[RG14a] Pablo Rauzy and Sylvain Guilley. A Formal Proof of Countermeasures Against Fault Injection Attacks on CRT-RSA. *Journal of Cryptographic Engineering*, pages 173–185, 2014.

[RG14b] Pablo Rauzy and Sylvain Guilley. Countermeasures Against High-Order Fault-Injection Attacks on CRT-RSA. In *IACR Workshop on Fault Diagnosis and Tolerance in Cryptography*, pages 68–82. IEEE, 2014.

[SAB10] Edward J. Schwartz, Thanassis Avgerinos, and David Brumley. All You Ever Wanted to Know About Dynamic Taint Analysis and Forward Symbolic Execution (but Might Have Been Afraid to Ask). In *Proceedings of the 2010 IEEE Symposium on Security and Privacy*, pages 317–331. IEEE, 2010.

[SGA07] Michael Sutton, Adam Greene, and Pedram Amini. *Fuzzing : brute force vulnerability discovery*. Pearson Education, 2007.

[SKW07] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago : a core of semantic knowledge. In *WWW*, pages 697–706. ACM, 2007.

[SL00] G.M. Swor and D.K. Lawrence. Health care information and data tracking system and method, 2000. US Patent 6,148,297.

[SMOS12] Nikos Sidiropoulos, Micha Mioduszewski, Pawe Oljasz, and Edwin Schaap. Open Wifi SSID Broadcast vulnerability, 2012.

[SSH+14] Alexander Schaub, Emmanuel Schneider, Alexandros Hollender, Vinicius Calasans, Laurent Jolie, Robin Touillon, Annelie Heuser, Sylvain Guilley, and Olivier Rioul. Attacking Suggest Boxes in Web Applications Over HTTPS Using Side-Channel Stochastic Algorithms. In *9th International Conference on Risks and Security of Internet and Systems*, pages 116–130. Springer Berlin Heidelberg, 2014.

[ST12] Fu Song and Tayssir Touili. Efficient malware detection using model-checking. In *FM 2012 : Formal Methods*, pages 418–433. Springer Berlin Heidelberg, 2012.

[ST13] Fu Song and Tayssir Touili. PoMMaDe : Pushdown Model-checking for Malware Detection. In *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, pages 607–610. ACM, 2013.

[VHBH13] Nevena Vratonjic, Kévin Huguenin, Vincent Bindschaedler, and Jean-Pierre Hubaux. How Others Compromise Your Location Privacy : The Case of Shared Public IPs at Hotspots. In *13th Privacy Enhancing Technologies Symposium (PETS)*, pages 123–142. Springer Berlin Heidelberg, 2013.

[VHBH14] Nevena Vratonjic, Kévin Huguenin, Vincent Bindschaedler, and Jean-Pierre Hubaux. A Location-Privacy Threat Stemming from the Use of Shared Public IP Addresses. *IEEE Transactions on Mobile Computing*, pages 1536–1233, 2014.

[VK14] Denny Vrandečić and Markus Krötzsch. Wikidata : a free collaborative knowledgebase. *Communications of the ACM*, pages 78–85, 2014.

## Enseignement

Ma mission doctorale d'enseignement a été l'occasion de me confronter pour de vrai à cette partie du métier d'enseignant-chercheur. En particulier au premier semestre de l'année 2014-2015 où j'ai été responsable d'une UE complète : incluant non seulement la conception, l'encadrement, et les corrections des cours magistraux, TD, TP, projets, examens papiers, et examens machines, mais aussi la partie administrative de la gestion d'une UE.

L'offre de formation en informatique de Paris 8 au département MIME et dans les autres composantes de l'UFR MITSIC est assez large, et beaucoup des cours proposés sur le site de l'UFR m'intéressent. Spécifiquement, le contexte d'un département informatique dans une université très majoritairement dédiée aux SHS est quelque chose qui m'attire beaucoup, en partie parce que le sujet de recherche auquel je m'intéresse au sein de la sécurité informatique, c'est à dire la protection de la vie privée, est à l'interface avec les SHS, mais aussi parce que l'offre de formation qui en découle me semble du coup très intéressante et originale.

En informatique, je suis très intéressé par l'enseignement des cours dit "de base". Je pense notamment aux cours de programmation dans différents paradigmes, aux cours de systèmes, de compilation, des technologies du web, ainsi qu'à ceux sur les fondements mathématiques de l'informatique (logique, mathématiques discrètes, calculabilité, complexité, etc). Le fait que la plateforme Racket soit utilisée pour le cours d'introduction à la programmation me rend aussi assez enthousiaste, étant moi-même un aficionado de Lisp et de Racket en particulier.

Au niveau des cours de master, je suis évidemment (de par mon profil et mon projet de recherche) intéressé par le parcours ISE, mais également par des cours d'autres parcours informatique (par exemple, le cours "Réseau & sécurité Web" du parcours THYP, ou le cours "Logiciels libres et protection de données" du parcours MIASHS).

En plus des formations strictement informatique, il y a la mention "Humanités numériques" du master qui me semble intéressante et dans laquelle il y a sûrement beaucoup de chose à faire en lien avec la protection de la vie privée, notamment dans le parcours NET.

En dehors de ce qui existe déjà, je suis assez motivé pour proposer à moyen terme des projets d'initiation à la recherche sur divers sujets, notamment en essayant de trouver des interfaces avec les sciences sociales sur les thématiques liées à la protection de la vie privée.

Lors de ma visite du LIASD et de l'université j'ai aussi appris l'existence du "Bocal". Un lieu de vie et de travail aussi accessible est quelque chose que je pense très important, aussi bien pour la formation (travail en groupe) que pour la vie sociale des étudiant·e·s, et qui n'est pourtant pas si courant dans les universités. Son existence fait parti des nombreuses petites choses qui ne sont pas directement liées à la fonction d'enseignant-chercheur mais qui me motivent aussi particulièrement à enseigner à Paris 8.

## Intégration

La recherche en sécurité et particulièrement son approche offensive est à double tranchant. La découverte d'une faille permet bien sûr de la colmater, mais facilite aussi son exploitation. L'éthique de la divulgation responsable est une partie de la solution à ce problème. La liberté académique, permise par le fait de faire ce type de recherche au sein d'un organisme public et indépendant comme le sont les universités, aide à se mettre à l'abri d'intérêts discordants dans ce genre de situation. Elle est aussi ce qui permet de s'intéresser sérieusement aux problèmes de respect et de protection de la vie privée, étant donnée l'immense valeur que prennent les données personnelles à l'heure de la publicité ciblée et du rançongiciel. En effet, les fournisseurs de services, tout comme les services gouvernementaux ne bénéficiant pas de la liberté académique, peuvent se montrer moins enclins à révéler au grand public dans quelle mesure leurs informations personnelles, et donc leur personne, sont vulnérables. C'est pourtant un droit des citoyens que d'en être informées, et un devoir de celles et ceux qui le savent de le leur dire. Je pense donc que le projet de recherche que je défends ici doit être mené dans une structure académique d'excellence bénéficiant d'une bonne visibilité internationale. L'Université Paris 8 est un établissement qui correspond idéalement à ce critère, et permettrait de mener à bien ce projet et par là-même de renforcer sa position d'interlocuteur privilégié en matière de sécurité et de protection de la vie privée.

### Au LIASD

En plus des informations sur le LIASD glanées en ligne, j'ai rencontré et discuté avec plusieurs membres du laboratoire avant de prendre la décision de candidater. Le LIASD est un laboratoire fortement multidisciplinaire, ce qui est particulièrement intéressant dans le cadre de mon projet de recherche qui présente aussi cet aspect là, d'un côté par son objectif de protection de la vie privée et de l'autre par son approche à la fois formelle et expérimentale.

Au LIASD lui même, la thématique de recherche "Manipulations, analyses et compréhension de programmes" entre en résonance avec une grande partie du premier axe de mon programme de recherche qui consiste en le développement de modèles et d'analyses de vulnérabilités. Dans cette partie de mon projet je compte en effet m'appuyer entre autre sur l'analyse dynamique de programme, et y contribuer en retour par l'hybridation de techniques d'analyses dynamique et statique.

Les deux thématiques "Médecine embarquée" et "Informatique des systèmes embarqués pour le handicap et la robotique" du LIASD me semble aussi permettre de nombreuses collaborations avec les membres du laboratoire. Les données médicales sont typiquement des données personnelles sensibles, et les systèmes embarqués étudiés dans ces thématiques posent très clairement des problématiques de sécurités. Cela en fait des cibles d'applications idéales de mon programme de recherche.

Lors de mes discussions avec certain·e·s membres du laboratoire, j'ai aussi appris que certaines des équipes de recherche de l'IUT de Montreuil qui dépend de Paris 8, sur le thème "Intégration de Réseaux d'Information hétérogènes et Interopérabilité" travaillent sur les ontologies (dans le cadre du web sémantique, mais pas seulement). Ce genre de travaux m'intéresse pour le second axe de mon programme de recherche, notamment pour réfléchir à l'évaluation des risques liés à l'inférence d'informations personnelles à partir d'un ensemble de données fixé.

Je sais aussi que la LIASD entretient déjà des contacts avec d'autres laboratoires et établissements, notamment avec l'EPFL, avec laquelle je compte aussi entrer en contact sur les sujets liés à la protection de la vie privée (notamment avec le groupe de recherche sur les systèmes décentralisés de Bryan Ford, qui s'intéresse de près à la thématique sécurité / privacy).

Enfin, la présence forte d'autres laboratoires sur les thématiques des sciences humaines et sociales est aussi intéressante dans le cadre des réflexions que je voudrais mener sur la formalisation de la vie privée (typiquement en interaction avec le droit mais aussi la sociologie et la psychologie).

### Collaborations externes

Je suis en contact avec David Naccache, qui a récemment monté une nouvelle équipe de sécurité de l'information au DI ENS et avec qui il pourrait être intéressant de collaborer dans le cadre de mon projet de recherche.

Je maintiens le contact avec l'équipe de Gilles Barthe à l'IMDEA Software Institute (à Madrid), pour

pouvoir continuer à collaborer sur les aspects liés à la preuve formelle de mon projet de recherche.

Le respect et la protection de la vie privée, en plus de leurs aspects informatiques, sont fortement liés à des domaines juridiques et sociaux. À ma connaissance seule l'équipe Inria Privactics de Claude Castelluccia et Daniel le Métayer englobe tous ces aspects de la privacy, et je resterai donc en contact avec eux pour collaborer quand cela semble nécessaire.

Les objectifs de la phase 3 de l'axe 2 de mon programme de recherche pourraient donner lieu à des projets en collaboration avec la CNIL, comme cela s'est déjà vu faire par exemple avec le projet Mobilitics [19].

19. `http://planete.inrialpes.fr/~achara/mobilitics/`