Pablo Rauzy

✉ *pablo.rauzyXXXinria.fr*
https://pablo.rauzy.name/

Curriculum vitæ – *candidat MdC 27ème*

Naissance : le 30 juillet 1989, à Marseille, nationalité française.
Adresse laboratoire : Laboratoire CITI (INSA Lyon / Inria),
56 bd Niels Bohr (CEI2),
69100 Villeurbanne.
Téléphone laboratoire : +33(0)4 XX XX XX XX
Adresse personnelle : XX XX XXXXXXX,
69100 Villeurbanne
Téléphone personnel : +33(0)6 XX XX XX XX

Parcours académique

2007 **Baccalauréat** série S option Sciences de l'ingénieur et spécialité Mathématiques, mention assez bien, lycée Marseilleveyre (Marseille).

2007 - 2009 **DEUG** Mathématiques et Informatique, mention félicitation du jury, Université de la Méditerranée (Marseille).

2009 - 2010 **Licence** Informatique, mention bien, École normale supérieure (Paris).
- *Stage L3* de 3 mois avec Christophe Rippert, Karin Altisen, et Kévin Marquet au Vérimag dans l'équipe Synchrone, mémoire intitulé "A formal approach to the development of system services in embedded systems".

2010 - 2012 **Master** Parisien de Recherche en Informatique, École normale supérieure (Paris).
- *Stage M1* de 5 mois avec Clemens Grelck à l'Université d'Amsterdam dans le Computer System Architecture group, mémoire intitulé "Implicit parallelization of code called from an external and already parallelized environment".
- *Stage M2* de 6 mois avec Marc Pouzet à l'ENS dans l'équipe Parkas, mémoire intitulé "Mixing continuous and discrete time in a synchronous language".

2009 - 2012 **Diplôme de l'ENS**, École normale supérieure (Paris). Département informatique.

oct 2012 - sept 2015 **Thèse** dans l'équipe SEN (Sécurité Électronique et Numérique) du département COMELEC (Communication et Électronique) de Télécom ParisTech (Paris).
- Titre : "Méthodes logicielles formelles pour la sécurité des implémentations de systèmes cryptographiques".
- Laboratoire : LTCI (UMR 5141)
- Directeur de thèse : Sylvain Guilley, Professeur, Télécom ParisTech.
- Soutenue le 13 juillet 2015 à l'École normale supérieure (mention très honorable).
- Composition du jury :
  - François Dupressoir, Chercheur, IMDEA Software Institute (examinateur),
  - Pierre-Alain Fouque, Professeur, Univ. Rennes 1 (rapporteur),
  - Karine Heydemann, Maître de Conférence, UPMC (examinatrice),
  - David Naccache, Professeur, Univ. Panthéon-Assas (examinateur - président),
  - Marie-Laure Potet, Professeure, Ensimag (rapporteuse),
  - Mehdi Tibouchi, Chercheur, NTT (examinateur),
  - David Vigilant, Chercheur, Gemalto (invité).
- Financement par une bourse de l'école doctorale ÉDITE (contrat doctoral + mission d'enseignement à Polytech'UPMC).
- *Stage* de 1 mois à l'IMDEA (Madrid) dans l'équipe de Gilles Barthe.
- *Co-encadrement* (à 50%) avec Sylvain Guilley du stage de M2 de Martin Moreau, sur la protection de calculs sur courbes elliptiques contre les attaques par injection de fautes ; ce stage a abouti à un article ainsi qu'à un chapitre d'ouvrage. Martin Moreau est maintenant en thèse à l'IMDEA avec Gilles Barthe.

oct 2015 - sept 2016 (situation actuelle) **Post-doc** dans l'équipe-projet Inria Privatics du laboratoire CITI commun à l'INSA Lyon et au centre Inria Grenoble–Rhône-Alpes (Lyon).
- Projet : "Formal Model for Privacy as Control".
- Encadrant : Daniel Le Métayer.
- Financement par l'équipe-projet Inria.

Domaine de recherche

Je travaille sur la *sécurité de l'information*, notamment sur le respect et la protection de la *vie privée*. D'une part, cela m'amène à m'intéresser à tous les niveaux de la sécurité : à la sécurité physique des implémentations de systèmes embarqués, à l'analyse de vulnérabilités des systèmes d'information et des réseaux, à la cryptologie, à la rétro-ingénierie de logiciels malveillants, et aux tests d'intrusion. D'une autre part, cela m'amène aussi à m'intéresser aux systèmes qui manipulent des données personnelles, en particulier aux systèmes distribués tels que les *réseaux intelligents* (smart grid, smart city, etc.), et de manière plus générale à l'*Internet des objets*.

Mon projet de recherche vise la protection de la vie privée par la mise en œuvre d'une *approche offensive* de la sécurité, c'est à dire de garantir cette dernière en cherchant systématiquement les vulnérabilités des systèmes. Pour cela je m'appuie sur une *approche formelle* de la preuve et de l'automatisation des méthodes de protection.

Production scientifique

Revues internationales à comité de lecture	2	Journal of Cryptographic Engineering (×2)
Chapitres d'ouvrage	1	publié chez CRC Press
Conférences internationales à comité de lecture	7	WISTP, IMACC, TGC, PPREW, FDTC, HOST, COSADE
Conférences internationales à comité de lecture sans acte	4	COSADE, PROOFS (×2), TRUDEVICE
Articles soumis ou en préparation	2	
Exposés lors de conférences internationales	6	dont 4 à l'étranger
Exposés lors de conférences nationales	1	à l'étranger
Séminaires	4	+ 3 invitations à venir
Présentations de posters lors de conférences internationales	2	dont 1 à l'étranger
Logiciels	3	dont 2 sont utilisés par des chercheurs et dans l'industrie

Responsabilités collectives

Membre des comités d'organisation des conférences COSADE 2013, COSADE 2014, et PROOFS 2015.
Relecteur pour les conférences COSADE 2014, SPACE 2015, et CARDIS 2015.
Relecteur externe pour le *Journal of Cryptographic Engineering* en 2013.

Enseignement

2012 - 2013
2013 - 2014 **Programmation en C** (32h + 32h) [N. Ouarti]. Encadrement des TP de langage C et programmation modulaire, à Polytech'UPMC (∼15 étudiant·e·s en 3ème année).

2012 - 2013
2013 - 2014 **Développement web** (32h + 40h) [H. Ouzia]. Encadrement des TP de HTML, CSS, PHP, et MySQL, à Polytech'UPMC (∼15 étudiant·e·s en 3ème année).

2013 - 2014 **Introduction à GNU/Linux et à la ligne de commande** (6h). Encadrement du "TP Zéro" de familiarisation avec GNU/Linux, à Polytech'UPMC (∼30 étudiant·e·s en 1ère année).

2014 - 2015 **Programmation orientée objet et langage C++** (54h). Responsable du cours entier (CM, TP, projets, examens machine et papier), à Polytech'UPMC (∼30 étudiant·e·s en 4ème année).

Mes activités d'enseignement sont décrites en détails par la suite.

Compétences techniques

Programmation	OCaml, C, Racket, C++, ASM	Scriptage	Bash, Python, Emacs Lisp
Web	HTML, CSS, JS, PHP, SQL	Graphisme	Inkscape, GIMP
Outils	LaTeX, Emacs, Git, SVN, Tor, I2P	Systèmes	UNIX, GNU/Linux

Langues

Français	langue maternelle	Anglais	lu / écrit / parlé

Mandats et affiliations

depuis 2008 Membre de l'April et de la FSF.

2008 - 2009 Élu représentant des étudiant·e·s en informatique au CÉVU de la Faculté de Sciences de Luminy.

2009 - 2012 Élu représentant des étudiant·e·s en informatique au conseil du Dpt informatique de l'ENS.

2011 - 2012 Élu représentant des étudiant·e·s scientifiques au Conseil d'administration de l'ENS.

depuis 2015 Co-fondateur et membre de l'association CAPSH qui porte le projet Dissemin (pour le développement du libre accès aux résultats de la recherche).

Pablo Rauzy

✉ pablo.rauzyXXXinria.fr
https://pablo.rauzy.name/

Publications et logiciels – *candidat MdC 27ème*

Les articles pour lesquels les noms de auteurs sont classés par ordre alphabétique plutôt que par ordre de contributions sont marqués d'un α.

Toutes mes publications sont disponibles en version intégrale depuis ma page web :
`https://pablo.rauzy.name/research.html#publications`

Revues internationales à comité de lecture

[RGN15] Pablo Rauzy, Sylvain Guilley, Zakaria Najm. **Formally Proved Security of Assembly Code Against Power Analysis**. *Journal of Cryptographic Engineering*, issue à paraître, 2015. DOI: 10.1007/s13389-015-0105-2.

[RG14a] Pablo Rauzy, Sylvain Guilley. **A Formal Proof of Countermeasures Against Fault Injection Attacks on CRT-RSA**. *Journal of Cryptographic Engineering*, Volume 4 Issue 3, 2014. DOI: 10.1007/s13389-013-0065-3.

Chapitres d'ouvrage

[RMG16] Pablo Rauzy, Martin Moreau, Sylvain Guilley. **Protecting Pairings-Based Cryptography Against Fault Injection Attacks**. Chapitre du livre *Handbook of Pairing Based Cryptography*, Nadia El Mrabet et Marc Joye éditeurs, CRC Press Taylor and Francis group, à paraître.

Conférences internationales à comité de lecture

[KKR+16] Ágnes Kiss, Juliane Krämer, Pablo Rauzy, Jean-Pierre Seifert. **Algorithmic Countermeasures Against Fault Attacks and Power Analysis for RSA-CRT**. *COSADE 2016: 7th International Conference on Constructive Side-Channel Analysis and Secure Design*. DOI: pas encore disponible. α

[RNR+15] Lionel Rivière, Zakaria Najm, Pablo Rauzy, Jean-Luc Danger, Julien Bringer, Laurent Sauvage. **High Precision Fault Injections on the Instruction Cache of ARMv7-M Architectures**. *HOST 2015: IEEE International Symposium on Hardware-Oriented Security and Trust*. DOI: 10.1109/HST.2015.7140238.

[RG14c] Pablo Rauzy, Sylvain Guilley. **Countermeasures Against High-Order Fault-Injection Attacks on CRT-RSA**. *FDTC 2014: 11th IACR Workshop on Fault Diagnosis and Tolerance in Cryptography*. DOI: 10.1109/FDTC.2014.17.

[RG14b] Pablo Rauzy, Sylvain Guilley. **Formal Analysis of CRT-RSA Vigilant's Countermeasure Against the BellCoRe Attack**. *PPREW 2014: 3rd SIGPLAN Program Protection and Reverse Engineering Workshop*. DOI: 10.1145/2556464.2556466.

[ABB+12] Antoine Amarilli, Fabrice Ben Hamouda, Florian Bourse, Robin Morisset, David Naccache, Pablo Rauzy. **From Rational Number Reconstruction to Set Reconciliation and File Synchronization**. *TGC 2012: 7th International Symposium on Trustworthy Global Computing*. DOI: 10.1007/978-3-642-41157-1_1. α (article invité)

[ANR+11] Antoine Amarilli, David Naccache, Pablo Rauzy, Emil Simion. **Can a Program Reverse-Engineer Itself ?**. *IMACC 2011: 13th IMA International Conference on Cryptography and Coding*. DOI: 10.1007/978-3-642-25516-8_1. α (article invité)

[AMN+11] Antoine Amarilli, Sascha Müller, David Naccache, Daniel Page, Pablo Rauzy, Michael Tunstall. **Can Code Polymorphism Limit Information Leakage ?**. *WISTP 2011: Workshop in Information Security Theory and Practice*. DOI: 10.1007/978-3-642-21040-2_1. α (article invité)

Les articles [AMN+11], [ANR+11], et [ABB+12] correspondent à des travaux effectués avant ma thèse.

Conférences internationales à comité de lecture sans acte

[RG15] Pablo Rauzy, Sylvain Guilley. **Towards Generic Countermeasures Against Fault Injection Attacks**. *TRUDEVICE 2015: 3rd Workshop on Trustworthy Manufacturing and Utilization of Secure Devices.*

[RGN14] Pablo Rauzy, Sylvain Guilley, Zakaria Najm. **Formally Proved Security of Assembly Code Against Power Analysis**. *PROOFS 2014: 3rd Workshop on Security Proofs for Embedded Systems.* Sélectionné pour soumission en version étendue au *Journal of Cryptographic Engineering.*

[RG13] Pablo Rauzy, Sylvain Guilley. **A Formal Proof of Countermeasures Against Fault Injection Attacks on CRT-RSA**. *PROOFS 2013: 2nd Workshop on Security Proofs for Embedded Systems.* Sélectionné pour soumission en version étendue au *Journal of Cryptographic Engineering.*

[RGD13] Pablo Rauzy, Sylvain Guilley, Jean-Luc Danger. **Software Countermeasures Against DPA Attacks: Masking vs Dual-Rail with Precharge Logic**. *COSADE 2013: 4th International Workshop on Constructive Side-Channel Analysis and Secure Design.* (article court)

Articles en soumission

[RGM+] Pablo Rauzy, Sylvain Guilley, Martin Moreau, Zakaria Najm. **Using Modular Extension to Provably Protect ECC Against Fault Attacks**.

Articles en préparation

[RL] Pablo Rauzy, Daniel Le Métayer. **Modeling Privacy as Control**.

Logiciels

paioli Cet outil permet de protéger du code assembleur contre les attaques par analyse de consommation de courant (comme la DPA ou la CPA) et de prouver formellement l'efficacité de la protection. Pour cela, il insère automatiquement une contre-mesure d'équilibrage connue sous le nom de DPL (*dual-rail with precharge logic*) dans du code assembleur "bitslicé" (typiquement sur un algorithme de chiffrement par bloc). Indépendemment, il est capable, en exécutant symboliquement le code un peu à la manière de l'interprétation abstraite, de vérifier statiquement si la consommation de courant d'un code assembleur donné est correctement équilibrée au regard d'un modèle de fuite (typiquement la distance de Hamming des mises à jour de valeurs, et le poids de Hamming des valeurs).

La création de cet outil, et les résultats obtenus avec, ont été publiés [RGN14,RGN15].

L'outil et ses résultats sont étudiés chez Gemalto. De plus, une version protégée de l'algorithme de chiffrement PRESENT qui a été obtenue avec paioli est actuellement utilisée et étudiée par le groupe Microsystems de NTU à Singapour.

J'ai intégralement développé cet outil. Code source et exemples disponibles sous licence CeCILL sur `https://pablo.rauzy.name/sensi/paioli.html`.

finja Cet outil permet l'analyse formelle de contre-mesures contre les attaques par injection de fautes de type BellCoRe sur les algorithmes de cryptographie asymétrique du type de CRT-RSA. Il utilise les règles de l'arithmétique modulaire pour faire de l'évaluation symbolique par réécriture. Cela permet une couverture complète des fautes possibles dans un modèle d'attaque donné (fautes aléatoires et/ou à zéro, permanentes et/ou transientes, nombre de fautes). On peut ainsi prouver la résistance d'une contre-mesure en fonction d'une condition de succès d'attaque. L'outil a aussi permis la simplification des codes et la diminution de la quantité de nombres aléatoires (coûteux à générer) nécessaires dans les contre-mesures de l'état de l'art, comme celles de Aumullër et al. (Infineon) et de Vigilant (Gemalto).

Les résultats obtenus avec cet outil ont donné lieu à trois publications [RG14a,RG14b,RG14c].

Cet outil a été réutilisé par Ágnes Kiss, Juliane Krämer, et Jean-Pierre Seifert de TU Darmstadt en Allemagne pour étudier formellement d'autres types de contre-mesures pour CRT-RSA. Leur travail a abouti à un article sur lequel nous avons collaboré [KKR+16].

J'ai intégralement développé cet outil. Code source et exemples disponibles sous licence CeCILL sur `https://pablo.rauzy.name/sensi/finja.html`.

enredo Cet outil permet de protéger les algorithmes de cryptographie asymétrique contre les attaques par injection de faute. Il applique une transformation de code prouvée correcte, qui insère la contre-mesure appelée *extension modulaire* qui permet à faible coût d'introduire une redondance dans le calcul, et ainsi de vérifier son intégrité.

Un article se basant sur les résultats de cet outil est en cours de soumission [RGM+], et un chapitre d'ouvrage [RMG16] va paraître.

La conception de cet outil a démarré lors de mon stage à l'IMDEA Software Institute avec Gilles Barthe.

J'ai intégralement développé cet outil. Code source et exemples disponibles sous licence CeCILL sur `https://pablo.rauzy.name/sensi/enredo.html`.

Pablo Rauzy

✉ pablo.rauzyXXXinria.fr
https://pablo.rauzy.name/

Travaux antérieurs – *candidat MdC 27ème*

Stages de recherche avant la thèse

Stage L3 :
3 mois dans l'**équipe Synchrone du laboratoire Verimag** à Grenoble avec Christophe Rippert, Karine Altisen et Kévin Marquet. Mémoire : "*une approche formelle du développement de services systèmes dans les systèmes embarqués temps réel*". Résultat principal : développement d'un ordonnanceur dynamique pour le langage Lustre qui conserve les garanties temps réel de l'ordonnancement statique.

Stage M1 :
5 mois dans le **groupe Computer Systems Architecture à l'université d'Amsterdam** avec Clemens Grelck. Mémoire : "*la parallélisation implicite de code appelé depuis un environnement extérieur et déjà parallélisé*". Résultat principal : redéveloppement de l'interface externe C du langage SaC (Single Assignment C, fait pour la parallélisation implicite de calculs sur des tableaux) pour lui permettre de faire de la parallélisation automatique même quand le code C qui s'interface avec le programme SaC contient déjà des threads, ce qui n'était pas possible avant.

Stage M2 :
6 mois dans l'**équipe Inria Parkas à l'ENS Ulm**, avec Marc Pouzet. Mémoire : "*le mélange de temps continu et discret dans un langage synchrone*". Résultat principal : une analyse statique de Zélus, un langage synchrone hybride (mixant temps discret et continu), qui détecte si une séquence infinie d'étapes discrètes peut arriver entre deux phases continues. La présence d'un nombre fini d'étapes discrètes entre deux phases continues est une condition nécessaire pour assurer que le temps avance pendant la simulation d'un système.

Thèse

Méthodes logicielles formelles pour la sécurité des implémentations cryptographiques

Les implémentations cryptographiques sont vulnérables aux attaques physiques, et ont donc besoin d'en être protégées. Une attaque physique exploite l'implémentation matérielle du système qu'elle cible. Elle peut utiliser des grandeurs physiques comme la consommation de courant du système, ou agir directement sur celui-ci, par exemple en fautant dans la mémoire physique des valeurs de variables intermédiaires du calcul. Bien sûr, des protections défectueuses sont inutiles. L'utilisation des méthodes formelles permet de développer des systèmes tout en garantissant leur conformité à des spécifications données. Le premier objectif de ma thèse, et son aspect novateur, est de montrer que les méthodes formelles peuvent être utilisées pour prouver non seulement les principes des contre-mesures dans le cadre d'un modèle, mais aussi leurs implémentations, étant donné que c'est là que les vulnérabilités physiques sont exploitées. Mon second objectif est la preuve et l'automatisation des techniques de protection elles-mêmes, car l'écriture manuelle de code est sujette à de nombreuses erreurs, particulièrement lorsqu'il s'agit de code de sécurité.

Les attaques physiques peuvent être classifiées en deux catégories distinctes. (1) Les attaques passives, où l'attaquant peut seulement lire l'information qui fuit par *canaux auxiliaires* (comme la consommation de courant ou les émanations électromagnétiques). Et (2) les attaques actives, où l'attaquant perturbe le système pour faire en sorte de lui faire révéler des secrets via sa sortie standard. Par conséquent, j'ai poursuivi mes objectifs dans ces deux cadres en considérant : (1) une contre-mesure qui diminue les fuites par canaux auxiliaires, et (2) des contre-mesures contre les attaques par *injection de faute*.

Il existe déjà des propriétés rigoureuses de sécurité pour les protections contre les fuites par canaux auxiliaires. Mes contributions se concentrent sur l'exploitation de méthodes formelles pour la conception et la vérification d'implémentations d'algorithmes protégés. J'ai développé une méthode de protection qui, étant donné une implémentation, en génère une version améliorée qui a un rapport signal à bruit nul sur ses canaux auxiliaires, grâce au fait que la fuite a été rendue constante (en particulier, la fuite ne dépend pas des données sensibles) en utilisant la méthode dite du "double rail" (*dual-rail with precharge logic*). Dans l'intérêt de la démonstration, j'ai aussi développé un outil (paioli) qui automatise l'application de cette méthode sur un code non sécurisé écrit en langage assembleur.

Indépendemment, l'outil permet de prouver que la propriété de fuite constante est toujours vérifiée pour une implémentation donnée, en fonction d'un modèle de fuite choisi (en général, le poids de Hamming des valeurs et la distance de Hamming des mises à jour de valeurs, car ces modélisations collent assez bien à ce qui est observé expérimentalement). D'une part, cela permet de vérifier systématiquement le résultat de la méthode de protection, et d'autre part cela sert aussi de test de non-régression de sécurité en cas d'optimisation manuelle du code obtenu. À ma connaissance, paioli est le premier outil permettant de protéger automatiquement une implémentation contre les fuites par canaux auxiliaires en équilibrant sa fuite de manière prouvable.

Le code d'une implémentation protégée de l'algorithme de chiffrement par bloc PRESENT qui a été obtenu grâce à paioli est aujourd'hui utilisé et étudié à la Nanyang Technological University de Singapour.

→ Les résultats de ce travail ont été présentés lors de la conférence internationale avec comité de lecture mais sans acte PROOFS 2014 (à Busan, en Corée) [RGN14] puis publiés dans une revue internationale avec comité de lecture [RGN15] (publication jointe au dossier).

À l'inverse, la définition même des objectifs de sécurité n'était pas clairement établie pour les attaques par injection de faute lorsque j'ai commencé ma thèse. Les propriétés de sécurité à prouver n'ayant même pas été formellement énoncées, beaucoup de contre-mesures ont été publiées sans preuve. C'est seulement lors de ma thèse que les "conditions de succès d'attaque" ont été introduites. En conséquence, la première question a été d'évaluer les contre-mesures existantes par rapport à ces conditions de succès d'attaque. À cette fin, j'ai développé une méthode, basée sur l'évaluation symbolique par réécriture d'arbre en suivant les règles de l'arithmétique modulaire, qui permet la couverture complète des fautes possibles (suivant un modèle de fautes très général) sur un algorithme (implémentant une contre-mesure) et le calcul de leurs effets. J'ai implémenté cette méthode dans un outil (finja), qui m'a permis de vérifier et prouver certaines contre-mesures, de retrouver des attaques connues, et aussi d'en découvrir de nouvelles.

Cet outil a été réutilisé par Ágnes Kiss, Juliane Krämer, et Jean-Pierre Seifert de TU Darmstadt en Allemagne pour étudier formellement d'autres types de contre-mesures pour CRT-RSA. Leur travail a abouti à la rédaction d'un un article commun que nous avons publié à la conférence internationale avec comité de lecture COSADE 2016 [KKR+16].

→ Les résultats de ce travail ont été présentés lors de la conférence internationale avec comité de lecture mais sans acte PROOFS 2013 (à Santa Barbara, aux États-Unis) [RG13] puis publiés dans une revue internationale avec comité de lecture [RG14a].

La seconde question portait sur la minimalité des contre-mesures. J'ai entre autres étudié en profondeur l'une des contre-mesures de l'état de l'art (développée chez Gemalto par David Vigilant). Le résultat de cette étude formelle utilisant finja a été la simplification drastique de la contre-mesure, aussi bien au niveau de la longueur du code que de la nécessité de nombres aléatoires (qui sont coûteux à générer), et ce sans affecter ses propriétés de sécurité. En effet, avec mes simplifications, la contre-mesure est passée de 9 à 3 vérifications d'invariants et ne nécessite plus qu'un seul nombre aléatoire, contre 5 auparavant. Ce travail a montré la valeur ajoutée de l'approche formelle par rapport à l'ingénierie par essais-erreurs qui a été jusqu'à présent la méthode principale de développement de contre-mesures.

→ Les résultats de ce travail ont été présentés et publiés dans dans la conférence internationale avec comité de lecture PPREW 2014 (à San Diego, aux États-Unis) [RG14b].

Les contre-mesures existantes revendiquent de protéger contre une ou parfois deux fautes. Cependant, des attaques utilisant plus de deux fautes ont vu le jour, aussi bien en pratique qu'en théorie. L'utilisation de finja m'a permis de découvrir que les contre-mesures se revendiquant du second ordre (résistantes à deux fautes) ne fonctionnaient pas dans le modèle de faute que j'avais défini (celui-ci étant plus général que ceux des auteurs de ces contre-mesures). La troisième question était alors de concevoir une nouvelle contre-mesure d'ordre supérieur, capable de résister à un nombre arbitraire de fautes. À son tour, la conception d'une nouvelle contre-mesure soulève la question de ce qui fait réellement fonctionner une contre-mesure. Pour tenter de répondre à cette question, j'ai classifié les contre-mesures existantes en essayant d'extraire les principes de protection des techniques employées. Ce travail de catégorisation m'a permis de comprendre l'essence d'une contre-mesure, et, en me basant dessus, de proposer une recette de conception de contre-mesure pouvant résister à un nombre arbitraire (mais fixé) de fautes.

→ Les résultats de ce travail ont été présentés et publiés dans la conférence internationale avec comité de lecture FDTC 2014 (à Busan, en Corée) [RG14c] (publication jointe au dossier).

J'ai aussi remarqué que toutes les contre-mesures que j'ai étudiées sont des variantes d'optimisation d'une même technique de base qui consiste à vérifier l'intégrité du calcul en utilisant une forme de redondance homomorphe. Cette technique est indépendante de l'algorithme auquel elle s'applique, et aussi de la condition de succès d'attaque, puisqu'elle repose entièrement sur des propriétés des structures mathématiques dans lesquelles se trouve les données sensibles, c'est-à-dire l'arithmétique modulaire. J'ai donc proposé une propriété de résistance face aux attaques par injection de faute qui dépasse la notion de condition de succès d'attaque. La quatrième question a été d'appliquer cette technique de protection à tous les calculs de cryptographie asymétrique, puisqu'ils travaillent tous sur des données mathématiques similairement structurées. Dans cette optique, j'ai développé une abstraction des calculs de cryptographie asymétrique qui permet d'appliquer simplement la méthode de protection par redondance. J'ai formellement défini cette méthode en définissant une transformation de code par réécriture que j'ai prouvée correcte. J'ai écrit un compilateur (enredo) qui automatise cette transformation et a permis d'obtenir des implémentations protégées d'algorithmes pour lesquels aucune contre-mesure n'a été publiée mais qui sont déjà victimes de nombreuses attaques par injections de fautes, comme les calculs de multiplications scalaire sur courbe elliptiques ou les algorithmes de couplage.

La mise en œuvre et l'étude pratique de ce travail théorique ont été réalisées conjointement avec Martin Moreau, dont j'ai encadré le stage de M2 autour de ce sujet.

→ Les résultats de ce travail ont été rédigés dans un article [RGM+] (publication jointe au dossier) ainsi que dans un chapitre d'ouvrage [RMG16].

Post-doctorat

Après ma thèse, j'ai voulu continuer à travailler en sécurité de l'information avec une approche formelle, tout en souhaitant explorer le sujet du respect et de la protection de la vie privée. C'est donc assez naturellement que j'ai contacté Daniel Le Métayer de l'équipe Inria Privatics, et que nous avons décidé de travailler ensemble sur la formalisation d'une façon de définir le respect de la vie privée.

Modèle formel du contrôle sur les données personnelles

Plutôt que le "droit d'être laissé tranquille", comme originellement défini par Samuel Warren et Louis Brandeis, le respect et la protection de la vie privée (*privacy*) sont de plus en plus vus, dans la société du tout numérique, comme le contrôle que peut exercer un individu sur ses données personnelles. La mode est d'ailleurs à l'inclusion d'exigences de respect et de protection de la vie privée dès les premières phases de conception d'un produit ou service, en suivant l'approche "*privacy by design*". Cependant, alors même que les notions de "*privacy as control*" et "*privacy by design*" sont omniprésentes dans la littérature, aucune définition claire de leur signification n'existe à ce jour. En conséquence ces notions peuvent être interprétées de différentes manières et il est difficile de rendre leur mise en pratique systématique ou mesurable. Par exemple, la littérature informatique en matière de vie privée se concentre principalement autour de deux sujets : les "policy languages" qui permettent d'exprimer des politiques de gestion des données personnelles, et le contrôle d'accès qui a développé des variantes comme les "role-based access control", le "purpose-based access control", ou encore le "risk-adaptive access control" pour mettre en œuvre la notion de "privacy as control", sans jamais vraiment définir ce que c'est réellement.

Il ressort cependant de la littérature, des pratiques, et de la loi (vue au travers des recommandations de mise en application données par la CNIL), que le contrôle d'un individu sur ses données personnelles s'organise autour de trois axes : l'*utilisation* de ses données personnelles, le *consentement* à l'utilisation de ses données personnelles par d'autres, et la *connaissance* qu'il ou elle a de l'utilisation qui est faite de ses données personnelles.

Je travaille donc en ce moment à la mise au point d'un modèle formel du contrôle se basant sur ces trois axes. Plus précisément, je suis en train de concevoir un langage de modélisation du contrôle des utilisateurs sur leurs données personnelles dans un système d'information. Le but étant de créer un outil qui prendrait en entrée la description d'un système dans ce langage, ainsi qu'un ensemble de critères définissant ce que serait un "contrôle satisfaisant", et procéderait à une vérification automatique de la validité de ces critères dans tous les états du système décrit. Le cas échéant, l'outil permettrait de tracer d'où vient la perte de contrôle pour pouvoir y pallier, ou du moins émettre des recommandations d'amélioration du système étudié, que ce soit au niveau de ses spécifications ou de moyens de contrôles externes.

Ce modèle formel et l'outil qui l'implémente, pourront servir de socle à la création d'une mesure du niveau de respect et de protection de la vie privée qu'offre un système donné, et donc de quantifier le risque de perte de contrôle, ou encore d'évaluer l'efficacité de méthodes de protection du contrôle. À son tour, cette mesure devrait permettre d'extraire des principes généraux de bonnes pratiques vis-à-vis du contrôle des utilisateurs sur leurs données personnelles, et donc de définir plus précisément la notion de "privacy by design" afin de rendre plus facile l'implémentation de produits ou services respectueux et protecteur de la vie privée.

Ce projet venant juste de démarrer (octobre 2015) il n'y a pas encore de résultat publié, mais un article est en cours de préparation [RL].

Encadrement d'un stage de M2 recherche

Pendant le dernier semestre de ma thèse j'ai eu l'opportunité de co-encadrer le stage de recherche de Martin Moreau, étudiant du M2 "Sécurité Fiabilité et Performance du Numérique" de l'UPMC.

Nous lui avons proposé de travailler sur la protection des algorithmes de cryptographie basés sur les courbes elliptiques (ECC) contre les attaques physiques par injection de fautes, en réutilisant la technique dite d'"extension modulaire" introduite par Shamir pour protéger CRT-RSA (c'est-à-dire RSA optimisé avec le théorème des reste chinois) contre ce même type d'attaques.

Le stage s'est extrêmement bien déroulé et a abouti à la rédaction d'un article en cours de soumission [RGM+], et d'un chapitre étendant les résultats obtenus sur l'ECC aux calculs de couplages, qui fera partie du livre *Handbook of Pairing Based Cryptography* (Nadia El Mrabet et Marc Joye éditeurs, CRC Press Taylor and Francis group) [RMG16].

Ce stage a été l'occasion pour Martin de confirmer son envie de poursuivre dans la recherche et il est maintenant en thèse avec Gilles Barthe à l'IMDEA Software Institute, à Madrid.

Collaborations internationales

Pendant ma thèse je suis allé travailler un mois à Madrid avec Gilles Barthe dans son groupe **Computer-Assisted Cryptography à l'IMDEA Software Institute**. J'ai collaboré avec eux sur le démarrage de mon travail de formalisation et d'automatisation (enredo) de la réécriture de code de calculs arithmétiques (type cryptographie asymétrique) pour la protection automatique contre les attaques par injections de fautes. C'est à la suite de ce travail que j'ai encadré moi-même un stagiaire de M2 (Martin Moreau, qui est maintenant en thèse avec Gilles Barthe à l'IMDEA) pour poursuivre cette idée et la mettre en pratique, ce qui a donné naissance à la rédaction d'un article [RGM+], ainsi qu'un chapitre d'ouvrage [RMG16].

J'ai aussi collaboré pendant ma thèse avec Ágnes Kiss et Juliane Krämer de la Technische Universität Darmstadt en Allemagne, à la rédaction d'un article [KKR+16] faisant suite à un travail qu'elles ont effectué en utilisant une des méthodes que j'ai développées pendant ma thèse ainsi que l'outil (finja) que j'ai écrit pour l'implémenter.

Dissémination

Exposés de conférence internationale avec comité de lecture

- Software Countermeasures Against DPA Attacks.
  COSADE 2013, Paris, France.
- A Formal Proof of Countermeasures Against Fault Injection Attacks on CRT-RSA.
  PROOFS 2013, Santa Barbara, États-Unis.
- Formal Analysis of CRT-RSA Vigilant's Countermeasure Against the BellCoRe Attack.
  PPREW 2014, San Diego, États-Unis.
- Countermeasures Against High-Order Fault-Injection Attacks on CRT-RSA.
  FDTC 2014, Busan, Corée.
- Formally Proved Security of Assembly Code Against Power Analysis.
  PROOFS 2014, Busan, Corée.
- Towards Generic Countermeasures Against Fault Injection Attacks.
  TRUDEVICE 2015, Grenoble, France.

Exposés de conférence nationale

— Towards Generic Countermeasures Against Fault Injection Attacks.
Itinerant Crypto Seminars (Crypto'Spain), Madrid, Espagne, 2015.

Exposés de séminaire de recherche

— Formally Proved Security of Automatically Protected Software Against Physical Attacks.
DigiCosme Working Group, Orsay, France, 2013.
— Formal Proofs of CRT-RSA Countermeasures Against the BellCoRe Attack.
Formal Methods and Security seminar of Inria and DGA, Rennes, France, 2014.
— Power Analysis Immunity by Offsetting Leakage Intensity.
GDR SoC-SiP Digital Security Day on Side-Channel Attacks, Paris, France, 2014.
— Protecting Against Fault Injection Attacks : from CRT-RSA to All Asymmetric Cryptography.
Séminaire SAS, Gardanne, France, 2015.
— A Formal Approach to Cryptosystems Implementation Security.
68nqrt, Rennes, France, 2016.

À venir :

— Protecting Against Fault Injection Attacks: from CRT-RSA to All Asymmetric Cryptography.
Séminaire du laboratoire Vérimag, Grenoble, France, 2016.
— Formal Modeling of Privacy as Control.
Séminaire informatique critique au LAAS, Toulouse, France, 2016.

Posters

— Formally Proved Security of Assembly Code Against Leakage.
CHES 2013, Santa Barbara, États-Unis.
— A Generic Countermeasure Against Fault Injection Attacks on Asymmetric Cryptography.
CHES 2015, Saint-Malo, France.

Autres implications dans la communauté scientifique

Organisation — Membre des comité d'organisation des conférences COSADE 2013, COSADE 2014, et PROOFS 2015.

Relecture — Relecteur pour les conférences COSADE 2014, SPACE 2015, et CARDIS 2015.
Relecteur externe pour le *Journal of Cryptographic Engineering* en 2013.

Libre accès — Je suis depuis plusieurs années impliqué dans le mouvement pour le libre accès aux résultats de la recherche scientifiques. J'ai rédigé une introduction au sujet qui a entre autre été publiée sur la une du Club Médiapart. J'ai organisé deux conférences, et donné six exposés. Je suis aussi un des co-fondateurs de l'association CAPSH (Comité pour l'Accessibilité aux Publications en Sciences et Humanités), qui est derrière la plateforme Dissemin (`http://dissem.in/`).

Plus de détails sont disponibles sur `https://pablo.rauzy.name/openaccess.html`.

Enseignement

Avant la thèse (total 18h)

Proposition et co-encadrement de projets de recherche des étudiant·e·s de L3 du cours "informatique scientifique par la pratique" de D. Naccache à l'ENS (3×2h d'encadrement pour chaque projet) :

Autonomic : créer une version fonctionnelle du jeu Nomic, qui permet aux joueurs de faire évoluer les règles du jeu, en se basant sur une construction de Quine permettant au programme de se réécrire lui-même.

Btrsync : créer une version améliorée de l'outil de synchronisation de dossier `rsync`, optimisée pour transférer le moins de données possible (en contrepartie de calculs plus conséquents au niveau de la source et de la destination de la synchronisation). Le projet a abouti à la rédaction d'un article qui a été présenté à la conférence TGC 2012 [ABB+12].

Paper laundry : créer un outil permettant de retirer des PDF d'articles de recherche les traces de pistages laissées par les maisons d'édition, pour cela étudier les possibilités de stéganographie dans des PDF et la possibilité de "normaliser" la structure d'un PDF.

Mission doctorale d'enseignement à Polytech'UPMC (total 196h)

2012 - 2013 **Programmation en C** (∼15 étudiant·e·s, 32h + 32h).

2013 - 2014 J'ai été en charge pendant deux ans (32h / an) des séances TP en groupes d'une quinzaine d'étudiant·e·s, pour le cours enseigné par M. Ouarti en 3ème année (équivalent L3) de la filière "Électronique et Informatique parcours Informatique Industrielle".

Les TPs ont couvert les bases de la programmation en C : structures de données, pointeurs, gestion de la mémoire, compilation modulaire (Makefile).

La matériel pour ce cours (fiches de TP) a été fourni par M. Ouarti.

2012 - 2013 **Développement web** (∼15 étudiant·e·s, 32h + 40h).

2013 - 2014 J'ai été en charge deux ans (32h puis 40h) des séances de TP en groupe d'une quinzaine d'étudiant·e·s, pour le cours enseigné par M. Ouzia en 3ème année (équivalent L3) de la filière "Agroalimentaire" (il s'agit d'un cours d'ouverture en informatique).

Les TPs ont couvert les bases du développement web et de la gestion de base de données en utilisant HTML, CSS, PHP et MySQL.

Le matériel pour ce cours (fiches de TP) a principalement été fourni par M. Ouzia, à l'exception de deux séances de TP pour lesquelles j'ai rédigé les sujets (consistant à leur faire faire en deux séances un clone de Twitter) qui continuent d'être utilisés depuis. Les sujets que j'ai rédigés sont disponibles sur `https://pablo.rauzy.name/teaching.html#epu-tpweb`.

2013 - 2014 **Introduction à GNU/Linux et à la ligne de commande** (∼30 étudiant·e·s, 6h).

Encadrement de deux groupes (3h par groupe) d'une trentaine d'étudiant·e·s arrivant à Polytech'UPMC pour leur "TP Zéro" de familiarisation avec l'environnement GNU/Linux des salles informatiques de l'établissement, et introduction à la ligne de commande.

2014 - 2015 **Programmation orientée objet et langage C++** (∼30 étudiant·e·s, 54h).

En troisième année de thèse j'ai demandé à être responsable d'un cours entier, par envie de découvrir tous les aspects de la gestion d'un cours. François Pécheux, responsable des cours d'informatique à Polytech'UPMC, a accepté de me confier le cours de programmation objet et C++ de 4ème année (équivalent M1) de la filière "Électronique et Informatique parcours Systèmes Embarqués" (une trentaine d'étudiant·e·s).
J'ai choisi de ne pas réutiliser le matériel des années précédentes et de proposer un cours entièrement neuf créé par mes soins. Comme son intitulé l'indique, ce cours couvre la programmation en langage C++ : différences avec le C, programmation orientée objet puis spécifiquement basée sur les classes, surcharge des opérateurs, héritages, templates, exceptions, la STL, puis une bibliothèque multimédia (la SFML). Les TPs ont servi àmettre en pratique les nouvelles notions de chaque séance de cours tout en réutilisant les notions vues auparavant. Pour évaluer les étudiant·e·s, je leur ai fait faire un projet de choix libre (en groupe de deux ou trois) avec la contrainte que ce soit un jeu graphique, un examen individuel sur machine (TP plus long et noté), ainsi qu'un examen sur papier. Tout le matériel de cours est disponible sur mon site à l'adresse `https://pablo.rauzy.name/teaching.html#epu-cpp`.