Curriculum Vitae Détaillé

Salwa SOUAF

Sommaire

Récapulatif

1.1 Formation

J'ai suivi un cursus de formation d'ingénieur en informatique et parallèlement un master recherche en informatique et sécurité. J'ai ensuite effectué un doctorat en informatique, que j'ai commencé aux États Unis et où j'ai passé la moitié de la durée de ma thèse. J'ai soutenu le 15 décembre 2020. Pour l'année 2020-2021, j'ai été attachée temporaire d'enseignement et de recherche à CentraleSupélec campus de Rennes. Enfin, je fais actuellement un Post-doctorat au sein de l'équipe Binsec du laboratoire LIST au CEA.

1.2 Enseignement

J'ai enseigné l'informatique durant la troisième année de mon doctorat à l'INSA Centre Val de Loire, puis comme ATER à CentraleSupélec campus de Rennes. J'ai eu ainsi l'occasion d'enseigner une grande diversité de domaines, à des niveaux variés (de la 1ère à la 3éme année cycle d'ingénieur).

Un rapport retour d'expérience sur l'enseignement des méthodes formelle a était publié:

- Salwa Souaf and Frédéric Loulergue. Experience report: Teaching code analysis and verification using frama-c. In Proceedings First Workshop on Applicable Formal Methods, AppFM@FM 2021, virtual, 23rd November 2021, volume 349 of EPTCS, pages 69–75, 2021.

1.3 Recherche

Mes recherches portent sur l'utilisation des méthodes formelles pour définir, vérifier et assurer la sécurité. Durant ma thèse, j'ai appliqué cela au Cloud en travaillant sur une solution de courtage qui permet aux utilisateurs Cloud désirant migrer leur infrastructure sur le Cloud de décrire leurs besoins fonctionnels et non-fonctionnels. Le courtier, en utilisant la logique du premier ordre et le *model checking*, vérifie la demande et trouve un placement adéquat dans un environnement multi-Cloud.

Deux publications sont issues de ces travaux, dans des conférences internationales à comité de lecture.

- Salwa Souaf, Pascal Berthomé, and Frédéric Loulergue. A Cloud brokerage solution: Formal methods meet security in Cloud federations. In *2018 International Conference on High Performance Computing & Simulation, HPCS 2018, Orléans, France, July 16-20, 2018*, pages 691–699. IEEE, 2018. `https://hal.archives-ouvertes.fr/hal-02317089`.

- Salwa Souaf and Frédéric Loulergue. A first step in the translation of Alloy to Coq. In *Formal Methods and Software Engineering - 21st International Conference on Formal Engineering Methods, ICFEM 2019, Shenzhen, China, November 5-9, 2019, Proceedings, volume 11852 of Lecture Notes in Computer Science*, pages 455–469. Springer, 2019.
  `https://hal.archives-ouvertes.fr/hal-02317118`.

Curriculum Vitae

2.1 Informations personnelles

Nom	:	**Salwa SOUAF**
Adresse Professionelle	:	CEA List - Nano-INNOV 2 Bd Thomas Gobert - 91120 Palaiseau, France
Tel	:	+33 (0)7 61 43 63 46
E-mail	:	sal.souaf@gmail.com
Web page	:	`https://salwasouaf.netlify.app/`
Date de naissance	:	24 Septembre 1993
Nationalité	:	Marocaine

2.2 Formation

Doctorat en Informatique 2017 - 2020

Equipe SDS, LIFO, INSA Centre Val de Loire, Bourges, France
Northern Arizona University, Flagstaff, USA de 2017 à 2019

- Titre : *Formal Methods Meet Security in a Cost Efficient Cloud Brokerage Solution*
- Abstract : Avec la demande croissante sur les ressources Cloud, vient l'argumentation de nombre de fournisseurs de services Cloud. Ce qui rend le processus de choir entre les différents fournisseurs et offres difficile pour les clients potentiels. Le courtier Cloud proposé, est une entité tierce qui intermédiaire la relation entre les clients et les fournisseurs Cloud. Le courtier guidera le client tout au long du processus d'intégration du Cloud. Notre courtier prend en considération les exigences fonctionnelles (c'est-à-dire la quantité et la description des ressources) et non fonctionnelles (c'est-à-dire les propriétés de sécurité) du client dés le premier contact. Après avoir reçu la description de la demande du client, notre courtier commence par vérifier sa cohérence et renvoie un contre-exemple en cas d'incohérences. Nous utilisons des méthodes formelles couplés avec de la programmation linéaire pour vérifier la consistance et trouver le placement approprié dans une fédération de Clouds. Après avoir communiquer l'emplacement trouver, si il existe un, au client, ce dernier décidera soit d'accepter cette offre ou pas. Une fois le courtier reçoit la confirmation du client le modèle sera prêt pour le déploiement.
- Date de la soutenance : 15 Décembre 2020
- Directeurs de thèse :

    - Frédéric Loulergue (Professeur, Université d'Orléans)
    - Pascal Berthomé (Professeur, INSA CVL)
- Rapporteurs:
    - Catherine Dubois (Professeur, ENSIIE)
    - Fabrice Mourlin (MCF HDR, UPEC)
- Président du Jury:
    - Benjamin Nguyen (Professeur, INSA CVL)
- Examinateurs:
    - Maryline Laurent (Professeur, Télécom SudParis)
    - Hélène Coullon (MCF, IMT Atlantique)
    - Bertrand Cambou (Professeur, Northern Arizona University)

Master 2 (Informatique nomade et sécurité informatique) 2016 - 2017
Université d'Orléans, Bourges, France
En paralléle avec ma cinquiéme année à l'INSA Centre Val de Loire

Diplôme d'Ingénieur en Sécurité et Technologies Informatiques 2015 - 2017
INSA Centre Val de Loire, Bourges, France
Contrat de double diplômation entre l'INSA et l'ENSA

Diplôme d'Ingénieur en Informatique 2011 - 2017
ENSA, Khouribga, Maroc

2.3 Emplois

Contrat post doctoral **2021 - Aujourd'hui**
Equipe BinSec, CEA List, Gif-sur-Yvette, France

Attaché Temporaire d'Enseignement et de Recherche (ATER) **2020 - 2021**
Equipe CIDRE, INRIA, CentraleSupélec, Rennes, France

Contrat doctoral 2017 - 2020
Equipe SDS, LIFO, INSA Centre Val de Loire, Bourges, France
Co-dirigé :
Août 2017 - Mai 2019 à *Northern Arizona University* par *Frédéric Loulergue*
Mai 2019 - Août 2020 à l'*INSA Centre Val de Loire* par *Pascal Berthomé*

2.4 Publications

---

[1] Salwa Souaf, Pascal Berthomé, and Frédéric Loulergue. A cloud brokerage solution: Formal methods meet security in cloud federations. In *2018 International Conference on High Performance Computing & Simulation, HPCS 2018, Orléans, France, July 16-20, 2018*, pages 691–699. IEEE, 2018. `https://hal.archives-ouvertes.fr/hal-02317089`.

[2] Salwa Souaf and Frédéric Loulergue. A first step in the translation of alloy to coq. In *Formal Methods and Software Engineering - 21st International Conference on Formal Engineering Methods, ICFEM 2019, Shenzhen, China, November 5-9, 2019, Proceedings*, volume 11852 of *Lecture Notes in Computer Science*, pages 455–469. Springer, 2019. `https://hal.archives-ouvertes.fr/hal-02317118`.

[3] Salwa Souaf and Frédéric Loulergue. Experience report: Teaching code analysis and verification using frama-c. In Mario Gleirscher, Jaco van de Pol, and Jim Woodcock, editors, *Proceedings First Workshop on Applicable Formal Methods, AppFM@FM 2021, virtual, 23rd November 2021*, volume 349 of *EPTCS*, pages 69–75, 2021.

2.5 Présentations scientifiques

Conférence	- A first step in the translation of Alloy to Coq, ICFEM, Shenzhen, China, 2019
Séminaire	- Formal methods meet security in a cost efficient Cloud brokerage solution, BinSec Day, CEA-LIST, Gif-sur-Yvette, France, 2021 - Formal methods meet security in a cost efficient Cloud brokerage solution, CIDRE, Rennes, France, 2020 - A security aware Cloud Brokerage Solution, LIFO, Bourges, France, 2019 - Formal methods meet security in a Cloud brokerage solution, Northern Arizona University, Flagstaff, USA, 2018
Poster Session	- Formal methods meet security in a Cloud brokerage solution, Northern Arizona University, Flagstaff, USA, 2017
Summer School	- Formal methods meet security in a Cloud brokerage solution, SILM (Security of software / hardware interfaces), Rennes, France, 2019

2.6 Formations scientifiques

Summer School	SILM (Security of software / hardware interfaces) 2019 à INRIA, Rennes
MOOC (Coursera)	Google Cloud Platform Fundamentals: Core Infrastructure
Conferences	USENIX 2018 ESORICS 2019 Journée Méthodes Formelles pour la Sécurité, GDR Sécurité Informatique 2020

Northern Arizona University	Nanotechnology for Cybersecurity: l'utilisation de fonctions PUF (Physically Unclonable Functions) dans le développement d'un gestionnaire de mots de passe robuste. Statistical Methods: l'application de méthodes statistiques à des problèmes du monde réel. Les sujets incluent la régression simple et multiple, l'ANOVA, la conception expérimentale et l'analyse de données catégoriques. On aborde la conception d'études, la collecte de données, l'analyse des informations et la synthèse des résultats. Data Mining and Machine Learning Research Rotation: où nous avons dû changer nos superviseurs et travailler avec un autre chercheur sur une thématique différente du nôtre. J'ai travaillé avec Dr. Fatemeh Afghah pour trouver un lien entre la théorie des jeux et le courtage Cloud. J'ai proposé d'utiliser les techniques du *Moving Target Defense* pour sécuriser les placements de machines virtuelles. Un sujet que j'aimerais vraiment revisiter et approfondir à l'avenir.

2.7 Compétences informatiques

Langages	Alloy, Coq, JAVA, C/C++, Python, PHP, PL/SQL, R, Ruby
Protocols & APIs	XML, JSON, SOAP, REST
Operating systems	Linux (Ubuntu, Debian, Kali), Microsoft Windows, MacOS
Sécurité informatique	Surveillance de la sécurité et détection des intrusions, sécurité du système d'exploitation

2.8 Autres informations

Langues	Trilingue : Arabe, Français et Anglais.
Loisirs	Sport (Tennis de Table, Crossfit), Littérature, Cuisine.

Enseignement

Statut	Année	Niveau*	Intitulé	Nature	Heures ETD
Doctorant	19-20	2A	Java avancé	Cours, TD	48
			Spécifications formelles	Cours, TD	26.5
		1A	Initiation au génie logiciel	TD	8
ATER	20-21	1A	Sécurité d'information	Cours	6
			Réseau et sécurité	TD, TP	30
			Coding week : jeu vidéo	Projet	12
		2A	Compilation	TP	18
			Modèles de données	TP	12
		3A	Détection d'intrusion	TD	6
			Projet d'option	Projet	12
			Concept et language : C++	TP	6
			Attaques en mémoire	TP	6
			Systèmes concurrents et répartis	Cours, TP	10.5
			Techniques avancées d'attaques	TP	6
			Sécurité OS	TP	15
			Dévelepoment web	Cours	4.5
		MS	Python	Cours, TP	9
			Sécurité OS	Cours	4.5
			Total	240	

* 1A, 2A et 3A Cycle d'ingénieur équivalant, resp. à L3, M1 et M2; MS : Master Spécialisé en Cyber Sécurité.

3.1 Détails des enseignements

Java avancé

Niveau : 2A Cycle d'ingénieur, nombre d'étudiants : 52, volume : 10h40 CM, 32h TD

Ce cours s'adresse à des étudiants ayant déjà eu un cours de base de programmation objet en Java. Il présente entre autre les aspects reliés à la JVM, les modules, la programmation concurrente et l'introspection.

J'étais responsable du cours. J'ai assuré toutes les séances de cours et travaux dirigés. J'ai organisé les examens et leurs corrections.

Spécifications Formelles

Niveau : 2A Cycle d'ingénieur, nombre d'étudiants : 23, volume : 10h40 CM, 10h40 TD

J'ai monté de zéro le cours de Spécifications Formelles : le but est d'initier les étudiants à la spécification de programmes via l'outil Frama-C. Les étudiants sont souvent en difficulté du fait de la rigueur et des capacités d'analyse nécessaires pour ce cours. J'ai donc choisi de structurer mon cours sous format d'un tutoriel et de me baser sur l'étude approfondie d'un cas d'étude : la spécification d'un module d'un OS IoT appelé Contiki. Cela permet aux étudiants d'avancer à leurs rythme et d'acquérir la tournure d'esprit nécessaire pour spécifier des programmes. Cette expérience m'a permis de prendre du recul, d'analyser les difficultés d'apprentissage des étudiants et de faire un choix pédagogique qui me semblait approprié.

Initiation au génie logiciel

Niveau : 1A Cycle d'ingénieur, nombre d'étudiants : 24, volume : 8h TD

Le but général de ce TD est la familiarisation avec les pratiques et l'écosystème logiciel typiques d'une entreprise dont l'activité principale est le développement, en se concentrant sur l'étape de tests. On présente une introduction générale au différents types de tests ainsi qu'une implémentation en utilisant le framework de test *JUnit*.

Sécurité d'information

Niveau : 1A Cycle d'ingénieur, nombre d'étudiants : 24, volume : 6h C

Ce cours représente une introduction aux notions de la sécurité d'information. Il donne les définitions de vulnérabilités et d'attaques. Il explique en détail des exemples d'exploitation de vulnérabilités, à savoir le **Buffer Over Flow**, **injection SQL** et **Cross-site Scripting (XSS)**.

Le TP lié à ce cours consiste dans l'introduction au contrôle d'accès Unix classique. Les compétences à acquérir sont:

- comprendre les méta-données de sécurité,
- ce familiariser avec un systéme d'éxploitation Unix et ses commandes,
- écrire des règles de contrôle d'accès.

Réseau et Sécurité

Niveau : 1A Cycle d'ingénieur, nombre d'étudiants : 28, volume : 15h TD, 15h TP

Les différents TDs et TPs assurés ont comme but l'acquisition des compétences suivantes:

- administration d'équipements réseau,
- conception et implémentation de protocoles de communication,
- mise en place d'un réseau privé virtuel (VPN),
- initiation à l'évaluation de la sécurité des applications web.

Ces TDs et TPs sont assurés en français et en anglais.

Coding Week : Jeu vidéo

Niveau : 1A Cycle d'ingénieur, Nnmbre d'étudiants : 60, volume : 12h

Ce projet se déroule sur deux semaines dédiées. Il consiste à programmer un jeu en réalité virtuelle.

Lors de la première semaine, les étudiants découvrent les outils Unity et langages nécessaires (Python, C#) pour réaliser une scène 3D et se déplacer dans celle-ci en réalité virtuelle.

La deuxième semaine consiste en une « game jam » dont l'objectif est de produire un petit projet en réalité virtuelle. Ce type d'événement est un concours en temps limité pour produire un jeu innovant dans un état d'avancement limité mais pour autant montrant l'originalité des mécanismes mis en œuvre.

Compilation

Niveau : 2A Cycle d'ingénieur, nombre d'étudiants : 29, volume : 18h TP

Ce TP se fait en deux partie. Dans la première partie les étudiants construiront un compilateur en OCaml pour un petit langage (pas de fonctions, pas de types) vers RISC-V. La deuxième partie consiste à ajouter des fonctionnalités au langage source pour le faire ressembler à du langage C. À l'issue de cette étape, le compilateur devrait pouvoir compiler un petit jeu "Space Invaders" dans qemu-riscv

Modèles de données

Niveau : 2A Cycle d'ingénieur, nombre d'étudiants : 29, volume : 12h TP

L'objectif du TP est de réaliser un logiciel de gestion de bibliothèque avec une interface graphique. L'interface proposera deux types de rôles: l'emprunteur normal et l'administrateur de la bibliothèque. Les fonctionnalités attendues sont: pouvoir emprunter un livre, consulter la liste des emprunteurs, la liste des livres empruntés, ou non empruntés, etc. Ce TP fait le lien entre tous les TPs fait avant et qui traitent: la modélisation UML, les base de données, développement orientée objet (Java, Kotlin), développement d'interfaces graphiques en Swing, et la modélisation et développement du contrôleur sous forme d'automates d'états finis.

Détection d'intrusion

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 36, volume : 6h TD

TD Snort

L'objectif de ce TD est de se familiariser avec les systèmes de détection d'intrusion. On présente et manipule Snort, un outil de détection d'intrusions réseau, orienté signatures. On détecte les attaques contre les systèmes :

- en capturant les paquets qui transitent sur le réseau à l'aide d'une interface en mode *promiscuous*, permettant de recevoir la totalité des paquets réseau (et non uniquement ceux adressés à l'interface d'écoute),
- en confrontant chacun des paquets à une base de signatures d'attaques.

TD Prelude

L'objectif de ce TD est de mettre en place et manipuler une plateforme de supervision de sécurité à l'aide du logiciel Prelude. Ce dernier est un outil de gestion d'alertes (SIEM, Security Information and Event Management). Cette plateforme rend les différents services attendus d'un outil SIEM, à savoir : détection d'attaques, collecte et formatage des alertes (ou événements de sécurité au sens large), authentification mutuelle des composants de la plateforme, transport des alertes dans un canal chiffré, corrélation, stockage et visualisation des événements.

Projet d'option

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 3, volume : 12h

J'ai proposé un sujet de projet recherche qui porte sur la simulation d'attaques dans une fédération de Cloud. Les objectifs de ce projet sont la compréhension des vulnérabilités de sécurité reliées à la virtualisation dans le Cloud Computing ainsi que la simulation d'une attaque qui exploite une de ces vulnérabilités à l'aide d'un outil de simulation, CloudSim. J'encadre un groupe de 3 étudiants avec l'enseignant-chercheur Guillaume Piolle.

Concepts des langages de programmation : C++

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 36, volume : 6h TP

Les TPs encadrés portent sur la manipulation du langage C++ et la maîtrise de ses spécificités.

Attaques en mémoire

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 36, volume : 6h TP

L'objectif du TP est d'étudier les attaques par débordement de buffer. Les étudiants jouent le rôle d'assistant technique dont la prestation consiste à :

- expliquer le fonctionnement de serveur, le composant vulnérable, en précisant le rôle et le fonctionnement de chaque fonction,
- identifier tous les éventuels problèmes de programmation, les fonctions qui manipulent des pointeurs devront faire l'objet d'un examen approfondi,
- analyser la trace réseau et expliquer le principe de l'attaque,
- identifier la vulnérabilité et proposer des correctifs,
- reproduire l'attaque en développant un programme ou un script *ad hoc*.

Systèmes concurrents et répartis

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 36, volume : 3h CM, 6h TP

On était trois intervenants dans ce cours. La partie dont je me suis chargé était la partie BigData. Le cours présente une introduction générale au concept du BigData et l'apprentissage automatique. Les objectifs des TPs qui l'accompagne sont la familiarisation avec le framework Spark et l'utilisation de ce dernier pour faire du Machine Learning. J'ai préparé le support de cours en se basant sur des anciennes ressources, que j'ai mis à jour pour qu'il soit cohérent avec les informations et les versions actuelles. En ce qui concerne les TPs, j'ai préparé des TPs interactifs et qui peuvent être jouer en distanciel, cause de la situation sanitaire. J'ai invité les étudiants à faire recours à la platforme d'analyse databricks pour palier les problèmes reliés au gestion d'environnements et installations que les étudiants rencontrent et qu'on ne peut malheureusement pas résoudre à distance.

Techniques avancées d'attaques

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 4, volume : 6h TP

Ce TP portera sur la présentation de quelques techniques avancées d'attaques.

Sécurité OS

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 36, volume : 15h TP

C'est un nouveau cours qui était introduit à la formation. J'ai participé à la phase de montage des différents cours avec les différents responsables, cela m'a permis d'avoir une meilleur idée sur la procédure de montage de cours et collaboration entre les différents enseignants. Pour les TPs que je vais assurer, j'ai travaillé avec les différents responsables afin de créer des sujets de TPs qui traitent et couvrent les mêmes thèmes que ceux présentés dans leurs cours. L'objectif de ces TPs est de manipuler un ensemble de composants d'UNIX pour l'administration et l'authentification, ainsi que se familiariser avec le module SELinux (Security-Enhanced Linux).

Développement Web

Niveau : 3A Cycle d'ingénieur, nombre d'étudiants : 36, volume : 3h CM

Ma partie du cours présente la programmation web coté serveur. Le but du cours est d'introduire les différents langages et framework de programmation coté serveur. J'ai préparé mon cours d'une façon à découper la séance sur deux. La première pour présenter et introduire les concepts aisi que de comparer les différents frameworks existant. La deuxième sous une format de tutoriel afin d'apprendre et se familiariser avec le framework PHP Symfony pour la création et gestion des applications web avec un accent sur les bonnes pratiques et la sécurité.

Python

Niveau : Master Spécialisé, nombre d'étudiants : 20, volume : 3h CM, 6h TP

Ce cours représente une introduction des notions de bases de Python. Il se déroule sous forme d'un tutoriel d'introduction à la programmation en Python.

Sécurité OS

Niveau : Master Spécialisé, nombre d'étudiants : 20, volume : 3h CM

Ce cours introduit les notions fondamentales de la sécurité informatique, son écosystème et fait quelques rappels techniques essentiels sur les systèmes GNU/Linux.

3.2 Approche enseignement

J'ai eu l'occasion d'enseigner des matières diverses entre la première et la troisième année du cycle ingénieur, j'ai réussi à monter et gérer des cours entiers ainsi qu'assurer une variété de TPs et TDs. Grâce à cette expérience, je suis en mesure de prendre des charges d'enseignement variées en informatique, et en programmation dans la plupart des langages répandus.

Je souhaite enseigner à différents niveaux ; mes diverses expériences, de l'initiation au master, me poussent à vouloir poursuivre une pratique variée de l'enseignement. Ainsi, je souhaite continuer à enseigner les bases de l'informatique à travers des cours d'initiation à la programmation, et la transmission de premières notions de sécurité informatique. J'aime enseigner les cours d'initiation, car j'aime encadrer les élèves et j'espère pouvoir les aider à acquérir des compétences fondamentales solides ; comme je souhaite pouvoir encourager des vocations scientifiques en participant à des enseignements plus avancés. Il serait particulièrement stimulant de pouvoir enseigner à un niveau master des cours coïncidant avec mes intérêts et thèmes de recherche, comme la sécurité informatique, la modélisation et la vérification formelle.

Je suis également prête à assumer et m'investir dans les responsabilités administratives relatives aux enseignements que je donnerai. Je suis très motivée pour participer à la vie des équipes d'enseignement et de recherche. J'ai le souhait d'être un acteur dynamique au fonctionnement de ces équipes. Je souhaite participer activement dans l'amélioration du montage des cours, de leur suivi mais aussi de leurs évaluations. Des cours et travaux pratiques plus attractifs et interactifs, avec une dimension concrète qui met en relation la théorie et la pratique avec une attention particulière sur la façon de travailler des étudiants, qui doit être la plus proche possible de celle qu'ils seront amenés à avoir en entreprise. Des évaluations enrichissantes, sous forme de projets dès que possible. D'autre part, je sais me rendre disponible pour le suivi des étudiants et accorde beaucoup d'importance à leur orientation.

Recherche

J'ai effectué mon doctorat de 2017 à 2020 au sein du LIFO dans l'équipe SDS, à l'INSA Centre Val de Loire sous la direction de Frédéric Loulergue et Pascal Berthomé. J'ai passé la premiére moitié de la durée de ma thèse aux Etats Unis à la Northern Arizona University. Ma thèse s'intitule *Formal Methods Meet Security in a Cost Efficient Cloud Brokerage Solution* et porte sur l'utilisation des méthodes formelles et de la programmation linéaire afin d'augmenter la sécurité lors de l'intégration du Cloud Computing.

4.1 Travaux de recherche

Mes travaux de recherche portent sur deux axes principaux: l'utilisation du courtage Cloud et des méthodes formelles pour la sécurité, et le recours à des assistants de preuve pour augmenter la confiance dans les modèles Alloy.

Contexte général

Le Cloud computing a prouvé son approche révolutionnaire en fournissant divers services informatiques au cours des dernières années. Ainsi, avec la demande croissante sur les ressources Cloud, vient l'augmentation du nombre de fournisseurs de services Cloud. Cette augmentation induit deux challenges majeurs à l'adoption du Cloud. Le premier est de choisir le meilleur fournisseur et la meilleure offre pour ses besoins. Le deuxième réside dans les appréhensions liées aux problèmes de sécurité Cloud. Les clients recherchent des solutions qui assureraient leur besoins personnalisés en terme de sécurité. Ces challenges ont motivé ma thèse. En effet, le but général de ma thèse était de trouver une solution qui assiste les clients dans l'intégration du Cloud tout en tenant compte de leurs besoins fonctionnels et non fonctionnels et en leur assurant un niveau de sécurité personnalisé.

Courtage et Sécurité Cloud

L'évolution du Cloud computing a conduit à l'émergence de nombreux aspects corrélés, à savoir:

- **Le courtage Cloud**: En fait, la migration vers des services Cloud peut être trop complexe à gérer pour les consommateurs potentiels du Cloud. Un consommateur peut du coup demander des services Cloud à un courtier Cloud, au lieu de rentrer dans une relation direct avec un fournisseur Cloud. D'après la définition donnée par le NIST, un courtier Cloud est :

  > Une entité qui gère l'utilisation, les performances et la livraison des services Cloud et négocie les relations entre les fournisseurs Cloud et les consommateurs Cloud.

  En général, les services fournis par un courtier Cloud peuvent être classés comme suit:

  - Intermédiation de services: un courtier Cloud renforce un certain service, proposé par un fournisseur de services Cloud, en améliorant des capacités spécifiques et en fournissant des services à valeur ajoutée au client. Cette amélioration peut inclure la gestion de l'accès aux services, la gestion de l'identité ou l'amélioration de la sécurité.
  - Agrégation de services: un courtier Cloud peut combiner plusieurs offres du Cloud computing en un ou plusieurs nouveaux services. Il assure l'intégration des données et des services et assure la sécurité des transferts de données entre le client et les différents fournisseurs de services Cloud.

- Arbitrage de service: L'arbitrage est similaire à l'agrégation sauf que les services combinés ne sont pas fixes. Un courtier d'arbitrage a la flexibilité de choisir les services à fournir au client auprès de plusieurs fournisseurs de services. L'objectif est d'optimiser les services fournis au client. Par exemple, il peut utiliser un service de notation de crédit pour mesurer et sélectionner un fournisseur avec le meilleur score.

- **Le Multi-Cloud**: L'architecture multi-Cloud fournit un environnement dans lequel les utilisateurs peuvent utiliser les ressources de plusieurs fournisseurs de Cloud pour créer des environnements Cloud en dehors de l'infrastructure interne traditionnelle. Bien qu'il soit compliqué de basculer entre les fournisseurs de Cloud pour effectuer des tâches, les fournisseurs de services de Cloud s'efforcent de rendre cela de plus en plus efficace.

  L'adoption d'une architecture multi-Cloud présente de nombreux avantages, pour citer quelques-uns:

  - Reprise après sinistre: il existe un risque important lors de l'utilisation des ressources d'un seul fournisseur de Cloud, d'une cyber-attaque interrompant toutes les opérations, laissant les utilisateurs finaux inaccessibles jusqu'à ce qu'elle soit résolue. L'architecture multi-Cloud, en revanche, peut rendre les services résilients contre de telles cyberattaques, car l'infrastructure du client est dispersée sur différents fournisseurs de Cloud.
  - Éviter le blocage des fournisseurs: la plate-forme multi-Cloud permet aux organisations de sélectionner les meilleurs services de différents fournisseurs Cloud, en les adaptant à leurs objectifs organisationnels, plutôt que d'avoir à modifier leurs processus métier pour s'adapter à la configuration d'un fournisseur spécifique.
  - Optimisation des coûts du Cloud: plusieurs fournisseurs de Cloud proposent des services à différents prix. En étant en mesure de choisir différents services de différents fournisseurs, les utilisateurs finaux peuvent optimiser le coût de leur architecture Cloud.
  - Faible latence: la latence est inhérente aux services Cloud fournis à partir de serveurs situés à des emplacements distants. Dans un environnement multi-Cloud, les utilisateurs peuvent déployer des centres de données dans plusieurs régions en fonction des emplacements nécessaires. Ceci est particulièrement utile pour les organisations mondiales qui ont besoin de diffuser des données dans des emplacements géographiquement disparates tout en maintenant une expérience utilisateur satisfaisante.

La solution proposée dans ma thèse intègre les aspects, d'assurance de la sécurité du Cloud, le courtage Cloud et le Multi-Cloud. Le courtier Cloud proposé est une entité tierce qui intermédie la relation entre les clients et les fournisseurs Cloud. Le courtier guidera le client tout au long du processus d'intégration du Cloud. Il prend en considération les exigences fonctionnelles (c'est-à-dire la quantité et la description des ressources) et non fonctionnelles (c'est-à-dire les propriétés de sécurité) du client dès le premier contact. Après avoir reçu la description de la demande du client, notre courtier commence par vérifier sa cohérence par rapport au expressions définis et que le modèle doit vérifier. Dans le cas d'incohérence de la demande, il met en évidence les éventuelles causes de celle-ci. Sinon, on passe à la deuxième phase de placement, où le courtier cherche un placement optimisant le coût total de la mise en place dans une fédération de Cloud créer par ses soins. Dans l'implémentation proposée j'utilise des méthodes formelles couplées avec de la programmation linéaire pour respectivement vérifier la cohérence et trouver le placement approprié dans une fédération de Clouds. L'outil fait une proposition de déploiement de l'architecture au client.

Modélisation et vérification de la demande du client pour cela j'utilise le model finder KodKod. Ce dernier est un moteur relationnel à usage général, ciblant des problèmes tels que l'analyse de conception, l'analyse de code, la génération de cas de test, la planification et la planification. Il s'agit d'un chercheur de modèles (c'est-à-dire un moteur qui recherche des modèles d'une formule dans un univers fini) fourni sous forme d'API Java, pour un langage de contraintes combinant logique du premier ordre, algèbre relationnelle, fermeture transitive.

La programmation linéaire est utilisée pour trouver une solution au problème d'optimisation de coût afin de trouver un placement adéquat pour l'architecture demandé par le client. La programmation linéaire est une technique mathématique pour générer et sélectionner la solution optimale ou la meilleure pour une fonction objective donnée. Techniquement, la programmation linéaire peut être formellement définie comme une méthode d'optimisation (c'est-à-dire de maximisation ou de minimisation) d'une fonction linéaire pour un certain nombre de contraintes énoncées sous la forme d'inégalités linéaires. Dans l'implémentation on se sert de GLPK (GNU Linear Programming Kit). C'est un logiciel libre et open-source, écrit suivant le standard ANSI C, qui permet de résoudre des problèmes d'optimisation linéaire de variables continues ou mixtes (discrètes et continues). Ce kit est composé d'un langage de modélisation GNU MathProg et d'une bibliothèque de fonctions C (GLPK) utilisant le solveur Glpsol.

Modélisation et Preuve

Alloy est un langage de modélisation formel, il a une syntaxe et une sémantique formelles. Alloy cible la spécification formelle des modèles de données orientés objet. Il peut être utilisé pour la modélisation des données en général, il est bon pour spécifier les objets de classes, les associations entre eux et les contraintes sur ces associations. Alloy dispose également d'un outil de vérification appelé Alloy Analyzer qui peut être utilisé pour analyser automatiquement les propriétés des modèles Alloy

Alloy est une méthode formelle légère car elle repose sur l'hypothèse "small scope" : l'examen de tous les petits cas est susceptible de produire des contre-exemples intéressants. Cependant, l'analyseur Alloy ne peut pas montrer l'absence d'erreurs. D'autres outils formels tels que les prouveurs de théorèmes interactifs Coq et Isabelle ont été utilisés pour fournir des garanties très solides sur les logiciels vérifiés, y compris un compilateur C et le noyau d'un système d'exploitation.

En utilisant le langage Alloy lors de l'implémentation d'une partie de la solution de courtage proposé dans ma thèse, je ne pouvais pas ignorer ses limitations. En effet, je ne peux pas nier qu'il est très utile d'utiliser des méthodes formelles légères, mais cela sera encore plus intéressant si on pouvait avoir une deuxième étape qui permettra de réellement prouver les propriétés voulues. En pratique, si on voulait utiliser un outil tel qu'Alloy dans un premier temps pour modéliser et vérifier des propriétés sur un système, puis utiliser un outil plus lourd comme les assistants de preuves dans un second temps, la formalisation effectuée en premier est perdue. Dans la solution proposée j'ai choisi d'utiliser l'assistant de preuve Coq.

L'assistant de preuve Coq développé à l'Institut français de recherche en informatique et en automatisation (INRIA) et lancé pour la première fois en 1989 est un prouveur de théorème interactif. L'environnement Coq permet la définition d'assertions mathématiques. Le compilateur Coq vérifie et aide à trouver des preuves de ces assertions. Coq est basé sur le calcul des constructions inductives, un l-calcul de type d'ordre supérieur. Coq et le calcul des constructions inductives sont basés sur la correspondance de Curry-Howard, c'est-à-dire qu'un type correspond à l'énoncé d'un théorème, et un programme à la preuve d'un théorème.

Pour accompagner la transition de Alloy vers Coq, nous proposons un traducteur des modèles Alloy vers du code Coq et un bibliothèque pour aider les utilisateurs dans le processus de rédaction de preuves. Le résultat de ce travail a été publié dans [2].

4.2 Projet personnel de recherche

Étant en post-Doctorat au sein d'une équipe dynamique je souhaite élargir mes champs de travail et me mettre devant de nouveaux défis.

Cours terme Actuellement je me lance dans la sécurisation de code. Je travail sur la conception et développement d'une méthode contre les attaques ROP. Je souhaite proposé une contremesure implémenter en software ainsi qu'en hardware, dont le coup et l'overhead est négligeable contre le niveau de protection assuré.

Long terme L'implémentation du courtier proposé peut être parallélisée et exécutée en différentes étapes. L'outil peut être divisé en différents composants. Ces derniers peuvent être développés dans le cadre de plusieurs travaux collaboratifs avec des experts dans les domaines nécessaires, je peux même

imaginer des sous parties développées dans le cadre de projets par des étudiants. Les fonctionnalités et collaborations potentiels des différents composants peuvent être résumées comme suit:

- **Relation client** : ce composant sera responsable de toutes les actions liées au client, à partir de la description de la demande, de la vérification de la cohérence, du retour d'un contre-exemple en cas d'anomalie, jusqu'à la présentation d'une éventuelle solution de placement. Pour améliorer la phase de vérification en utilisant des méthodes formelles, je continue d'affiner les définitions des relations sécurités ainsi que les prouver.

- **Provisioning and management** : ce composant sera l'interface entre le courtier et les fournisseurs de ressources Cloud. Il sera responsable de la réservation des ressources, initial et ultérieur. Plusieurs algorithmes déjà proposés en academia pour faire les calculs des quantités de ressources nécessaires de départ, des ressources devant être provisionnées ou libérées à un moment donné en tenant compte du nombre de clients, de leurs besoins, et la tendance générale du marché. C'est une piste que je n'ai pas encore parcouru, mais je souhaite proposé aux chercheurs qui ont effectués ces travaux la possibilité de collaborer, implémenter leurs algorithmes et les intégrer dans ce composant.

- **Placement et configuration** : ce composant trouvera les ressources Cloud correspondantes qui satisfont les exigences fonctionnelles du client et il fera la configuration réseau adéquate pour satisfaire les exigences non fonctionnelles. En cas de ressources insuffisantes, il alerte le composant *Provisioning and management*. Pour ce composant je veux en premier temps améliorer le modèle linéaire proposé dans ma thèse et du coup continuer ma collaboration avec Prof. Pascal Berthomé afin de faire cela. Ce composant peut faire objet de discussion avec différents chercheurs intéressés par l'axe *Modélisation de systèmes concurrents, mobiles et/ou multi-agents.*

- **Surveillance de la fédération** : ce composant permettra au courtier de surveiller sa fédération (ressource fédérées) à la fois d'un point de vue fonctionnel et sécuritaire. Il peut aussi proposer une manière formelle pour vérifier l'état du système global à n'importe quel moment. Ainsi que donner aux clients une garantie de confiance que la sécurité de leurs architectures est maintenue dans le temps. Plusieurs collaborations soit locaux soit internationaux peuvent être imaginer pour l'implémentation de ce composant.