Raounak Benabidallah

Formation

2020 **Docteure en informatique**, *Laboratoire IRISA*, Université Bretagne Sud.

2016 **Master en Systèmes Informatiques Intelligents**, *Université des sciences et de la technologie Houari Boumediene*, Algérie.

2014 **Licence en Informatique académique**, *Université des sciences et de la technologie Houarie Boumediene*, Algérie.

2011 **Baccalauréat série sciences exactes avec mention Très Bien**, *Lycée Mohammed Hadjres*, Algérie.

Expérience professionnelle

Depuis janvier 2022 **Chercheuse en post-doctorat**, *CEA List*, Équipe de recherche Binsec.

2020-2021 **Enseignante-chercheuse LRU**, *Université de Rennes 1*, Laboratoire IRISA, Équipe de recherche DiverSE.

2016-2020 **Doctorante**, *Laboratoire IRISA. Équipe de recherche Archware*, Université Bretagne Sud, Vannes.
Titre : Identification des vulnérabilités de sécurité dans les systèmes logiciels
Dirigée par : Salah Sadou et Isabelle Borne
La thèse a pour objectif la détection des **vulnérabilités** dans le **code logiciel** en utilisant des méthodes d'**apprentissage automatique**

2016 **6 mois de stage interne**, Université des sciences et de la technologie Houari Boumediene, Algérie.
Visualisation d'un système d'information géographique (SIG).

2014 **4 mois de stage interne**, Université des sciences et de la technologie Houari Boumediene, Algérie.
Découverte de voisinage dans les VANETs (Réseaux véhiculaires).

5 rue de l'effort mutuel – 91120 Palaiseau, France
✉ *raounak.benabidallah@cea.fr*

Recherche

1. Contexte et travaux de thèse

J'ai réalisé ma thèse, intitulée "*Identification automatique des vulnérabilités de sécurité dans les systèmes logiciels*", dans le laboratoire de recherche IRISA, site de Vannes, dans l'équipe de recherche Archware. Mes travaux de recherche s'inscrivent dans l'intersection de trois domaines de recherche, à savoir le génie logiciel, la sécurité informatique et l'intelligence artificielle (IA). Plus particulièrement, j'ai utilisé des techniques de IA pour détecter des vulnérabilités de sécurité dans les logiciels informatiques. En effet, la menace posée par les vulnérabilités logicielles croît de manière exponentielle. Ce phénomène est dû, d'une part, à l'omniprésence des logiciels, et d'autre part, au nombre important de failles existantes. Pour faire face à ce problème, plusieurs stratégies ont été élaborées au fil du temps. Certaines visent à mettre en place de bonnes pratiques de développement et les intégrer dès la phase de conception tandis que d'autres consistent à effectuer des inspections de sécurité en indiquant les zones vulnérables. Cette thèse s'inscrit dans la deuxième catégorie de travaux et porte essentiellement sur la construction de modèles de prédiction de vulnérabilités. La création de ces derniers soulève différents problèmes. Le plus important étant le manque de données sur les vulnérabilités logicielles. À cet effet, dans cette thèse, je mets en place une chaîne de traitement complète allant de la création et l'annotation automatique d'un corpus de sécurité jusqu'à la construction et l'évaluation des modèles de prédiction de vulnérabilités. La première contribution de cette thèse est plus axée sur l'approche de construction de corpus que sur le corpus lui-même. L'approche est basée sur la conception de méta-scanners de vulnérabilités permettant d'identifier des vulnérabilités de code efficacement. Cela consiste à combiner plusieurs outils d'analyse statique en se basant sur leurs performances individuelles pour chaque catégorie de vulnérabilités. Ma deuxième contribution correspond au corpus *SecureQualitas* qui consiste en un corpus d'applications Java annotées avec les vulnérabilités qu'elles contiennent. J'ai créé ce corpus en utilisant un méta-scanner construit à l'aide de trois outils d'analyse de vulnérabilités. Enfin, ma troisième contribution est de construire un modèle de prédiction du code vulnérable. J'ai opté pour l'utilisation de métriques de qualité pour caractériser le code et j'ai étudié les performances des modèles à la fois sur des catégories de vulnérabilités apprises par les modèles et sur des catégories non encore connues par le modèle. Les résultats de mes expérimentations ont montré l'efficacité des modèles sur les deux populations de vulnérabilités : connues et non connues.

Mots-clés: Analyse statique, identification de vulnérabilités, apprentissage automatique, modèles de prédiction.

Composition du jury de thèse:

Présidente	Christelle URTADO	HDR à l'école des mines Alès
Rapporteurs/	Tegawendé F. BISSYANDE	Professeur à l'Université du Luxembourg
Examinateurs	Chouki TIBERMACINE	HDR à l'université de Montpellier
Directeur	Salah SADOU	Professeurs à l'université Bretagne Sud
Co.directeur	Isabelle BORNE	Professeure à l'université Bretagne Sud

2. Collaborations

En plus de ma thèse, j'ai eu l'opportunité de collaborer avec d'autres chercheurs qui s'intéressent à diverses thématiques de recherche. Ces collaborations ont été très bénéfiques pour mon début de carrière car elles m'ont permis de m'ouvrir à de nouvelles idées et à développer un esprit critique. Les principales collaborations sont avec :

- Professeur **Eric Coatanea** de l'université de Tampere (Finland) :
  - Contact : eric.coatanea@tuni.fi
  - Description : je travaille avec Eric depuis deux ans sur le problème d'**identification de contradictions** dans la spécification de la sécurité. Eric vient du domaine de l'**ingénierie des systèmes** et a développé des méthodes très intéressantes pour détecter des contradictions dans des spécifications industrielles. Plus concrètement, une des méthodes développée par Eric transforme tout problème donné en un graphe où les noeuds représentent des composantes du système et les arcs les relations entre ces dernières. À chaque type de relation, un ou plusieurs effets sont associés selon le type de composantes. En utilisant les effets relatifs aux relations et les données en entrée pour chaque noeud, il est possible de détecter des contradictions de spécifications et de les corriger.
    Dans le domaine de la sécurité, les concepteurs sont souvent amenés à définir les composants à risque qu'il faut impérativement protéger. Ces composants sont appelés "*assets*". Afin de sécuriser au mieux ces assets, les concepteurs déterminent, pour chaque asset, la(les) propriété(s) de sécurité importante(s) à maintenir. Cependant, les assets sont souvent reliées entre elles et avec d'autres composantes du système et les relations peuvent introduire des effets non désirables pour la sécurité du système. À cet effet, nous tentons d'aider les concepteurs en pointant les zones à risque dans leur spécification. Pour ce faire, notre équipe, constituée de Salah Sadou, Eric Coatanea, Nan Messe et moi même, a étudié toutes les propriétés de sécurité existantes et les éventuels effets possibles entre elles. Afin d'éliminer la subjectivité, nous exploitons la base CVE qui contient un nombre important de vulnérabilités exploitées avec les propriétés de sécurité associées. Nous en tirons les effets entre les propriétés que nous les projetons sur un graphe représentant le système à sécuriser pour détecter les éventuelles contradictions. Un deuxième travail sera entamé par la suite utilisant la méthode soviétique TRIZ afin de proposer des solutions lorsqu'une contradiction est détectée.

- Professeur **Tegawandé Bissyandé** de l'Université du Luxembourg :
  - Contact : tegawende.bissyande@uni.lu
  - Description : ma collaboration avec Tegawendé a débuté en novembre 2019 où j'ai effectué un **séjour scientifique** dans l'équipe SERVAL au laboratoire SnT (Luxembourg). Tegawendé est la personne avec qui j'ai le plus échangé durant mon séjour au Luxembourg. Il travaillait notamment sur la correction des bugs (défauts) logiciel et la réparation de code. La **réparation automatique** des programmes est un nouveau domaine de rectification automatisée particulièrement utilisé pour corriger des bugs logiciels ou des erreurs de programmation. Parmi les solutions proposées dans ce domaine, nous retrouvons celle basée sur la génération automatique de patchs correctifs. L'objectif est de trouver une modification (minimale) d'un programme défectueux P pour produire un programme P' débarrassé du défaut. L'idée de base est d'abstraire, à partir de plusieurs exemples de changements de code, un modèle qui peut être appliqué à un code défectueux pour corriger la faille. Cette méthode produit des résultats prometteurs dans le cadre de la correction de défauts.
    Les vulnérabilités sont classées parmi les défauts logiciels les plus critiques qui existent. Lorsqu'elles sont identifiées, les programmeurs tentent également de produire des correctifs qui permettent de remédier à la vulnérabilité le plus rapidement possible. Ce problème nécessite de recourir à des méthodes de réparation automatique des programmes pour générer automatiquement des correctifs de vulnérabilités. Dans ce projet de collaboration, nous nous inspirons des travaux existants sur la réparation automatique de programmes afin de générer des patterns permettant la correction des vulnérabilités logicielles. Pour ce faire, la première étape consiste à collecter un ensemble de patchs qui ont permis la correction d'une vulnérabilité. Pour déterminer si un changement de code correspond à une éventuelle correction, il existe différentes méthodes telles que l'exploitation de bases de vulnérabilités existantes, l'utilisation d'outils d'analyse statique ou encore la récupération des correctifs GitHub. À partir d'un ensemble de patchs corrigeant une vulnérabilité, il est possible d'abstraire un pattern correctif, qui appliqué à un code, permet de corriger cette vulnérabilité.

3. Encadrement

- Stage IUT (Bac+2) (2018)
  - Durée : 10 semaines
  - Public : un étudiant en deuxième année DUT
  - Description : ce projet avait pour objectif d'initier le candidat à la sécurité informatique et au développement logiciel. Dans le cadre de ma thèse, j'ai construit un modèle de prédiction permettant de détecter les vulnérabilités logicielles. J'ai donc proposé ce projet pour créer un programme qui intègre ce modèle de prédiction à l'IDE de développement et qui permet d'alarmer le développeur dès qu'une vulnérabilité est ajoutée. Le développeur peut analyser l'alerte et corriger la vulnérabilité, ou bien signaler une fausse alerte. Afin d'améliorer le modèle, le programme récupère les résultats d'analyse du développeur et relance l'apprentissage après plusieurs modifications de la base d'apprentissage.

- Projet tutoré : initiation aux vulnérabilités logicielles et aux différents outils d'analyse statique (2018)
  - Durée : 4 mois
  - Public : 4 étudiants ingénieurs en 2ème année ENSIBS
  - Description : l'objectif de ce projet était d'approfondir les connaissances des étudiants dans le domaine de la sécurité logicielle. Pour ce faire, la première partie du projet était consacrée aux vulnérabilités logicielles : comprendre la définition des vulnérabilités et essayer de les identifier. Plusieurs outils ont été testés par les étudiants, ce qui nous a permis de créer une base conséquente de vulnérabilités logicielles

- Projet tutoré : correction de vulnérabilités à partir d'exemples de correctifs (2020)
  - Durée : 4 mois
  - Public : 4 étudiants ingénieurs en 2ème année ENSIBS
  - Description : tout comme le projet précédent, celui-ci s'inscrit dans le cadre de la sécurité logicielle. Cependant, l'objectif de ce projet est de proposer des corrections automatiques aux vulnérabilités en utilisant des techniques de IA. Les étudiants se sont principalement focalisés sur la compréhension du problème et sur les méthodes de création de bases de patchs. La solution adoptée consistait à utiliser un outil d'analyse de vulnérabilité et de localiser les changements d'état de code : vulnérable $=>$ non vulnérable.

- Stage ingénieur :
  - Durée : 4 semaines
  - Public : un étudiant ingénieur en 2ème année ENSIBS
  - Description : ce projet est la continuité du projet précédent avec un seul étudiant.

- Stage recherche M2 : correction automatique de vulnérabilités (**en cours**)
  - Durée : 6 mois
  - Public : un étudiant en 3ème année ENSIBS
  - Équipe : Salah Sadou, Tegawendé Bissyandé et moi même
  - Description : ce projet a pour objectif de créer des correctifs (patchs) de vulnérabilités. Contrairement aux projets précédents, celui-ci se focalise sur les techniques d'IA permettant de créer des patterns de corrections de vulnérabilités.

4. Visibilité

Durant ma dernière année de thèse, j'ai réalisé un séjour scientifique au laboratoire SnT au Luxembourg. J'ai été accueillie dans l'équipe Serval par Yves Le Traon (yves.letraon@uni.lu) et Tegawendé Bissyandé (tegawende.bissyande@uni.lu). Durant ce séjour, j'ai pu échanger avec plusieurs membres de l'équipe sur des thématiques similaires à la mienne ou complémentaires. J'ai notamment pu commencer une collaboration avec Tegawendé sur son projet sur la correction de vulnérabilités.

5. Projet de recherche

Mon expérience en recherche m'oriente fortement vers des thématiques liées à la sécurité informatique de façon générale, et à la sécurité logicielle en particulier. Je suis particulièrement intéressée par la détection et la correction de vulnérabilités afin de prévenir les attaques de sécurité. Les méthodes utilisées peuvent varier : de la modélisation, à la représentation en graphes, notamment les graphes d'attaque.
Sur le court terme, j'envisage de finaliser mes travaux de thèse en intégrant de nouvelles expérimentations sur l'ingénierie des caractéristiques et ce dans le but d'améliorer les performances de mes modèles de prédiction. L'idée est de ne prendre en considération que les métriques qui ont un réel impacte sur l'existence de vulnérabilités. Durant ma thèse, j'ai étudié plusieurs métriques logicielles mais je souhaiterais intégrer de nouvelles catégories de métriques afin d'avoir plusieurs vues des données en entrée. Aussi, il serait intéressant d'étudier les techniques de text minig dans le cadre de l'identification de vulnérabilités. Outre la détection de vulnérabilité, je suis vivement intéressée par la correction de celles-ci. Je considère ce projet comme une continuité logique de mes travaux de thèse car c'est le seul moyen de sécuriser les systèmes informatiques. Je m'intéresse aux méthodes d'apprentissage automatique et profond qui peuvent être appliquées pour proposer des solutions aux développeurs. De plus, les techniques de NLP (Natural Language Processing) sont inévitables pour la réussite de ce travail. Je suis aussi intéressé par un domaine nouveau pour moi qui relie la sécurité et le machine learning. Il s'agit d'identifier des exemples contradictoires "adversarial examples" qui trompent les modèles de prédiction et qui influent sur la sécurité des systèmes.

Enseignement

1. Modules enseignés

Durant ma thèse, j'ai eu l'occasion d'intervenir en tant qu'enseignante vacataire à l'Université Bretagne Sud de 2017 à 2019. Plus particulièrement, j'ai enseigné différentes matières à la Faculté des Sciences de l'Ingénieur, en licence Informatique. Ces matières comportaient à la fois une partie théorique et une partie pratique. Cette dernière représentait un grand pourcentage de ma charge horaire. Le défi principal de ces enseignements est de gérer des classes très hétérogènes.

2017-2019 **Enseignante vacataire à l'Université Bretagne Sud**.

- **Compréhension des systèmes informatiques**
  - Public : L1 licence Informatique
  - Volume horaire : 28h
  - Responsable de l'unité d'enseignement : Yves Mahéo (Yves.Maheo@univ-ubs.fr) Description : ce cours est destiné aux étudiants de première année en licence informatique et sert d'introduction à plusieurs systèmes informatiques. Ce cours est composé de trois

volets importants, à savoir l'imagerie ou données multimédia plus généralement, l'étude de réseaux sous Linux et enfin la gestion de systèmes d'exploitation (principalement du Shell).
  - Rôles : intervention en TP, correction de rendus de TP.

- **Algorithmique et programmation impérative (Programmation Python)**
  - Public : L1 licence Informatique
  - Volume horaire : 45h
  - Responsable de l'unité d'enseignement : Sylvie Gibet (sylvie.gibet@univ-ubs.fr) Description : ce cours a pour objectif d'introduire les notions fondamentales de programmation impérative aux étudiants de première année informatique. Ces notions sont mises en pratique en utilisant le langage Python où les étudiants apprennent à rédiger des programmes et de créer des interfaces graphiques. Les algorithmes de tri (par sélection, insertion, fusion, etc) sont présentés ainsi que la notion de complexité de programmes.
  - Rôles : intervention en TP, surveillance et correction des examens, correction des rendus de TP.

- **Logique et base de données**
  - Public : L1 licence Informatique
  - Volume horaire : 22h
  - Responsable de l'unité d'enseignement : Nicolas Courty (nicolas.courty@univ-ubs.fr)
  - Description : ce cours est divisé en deux parties : la première vise à introduire différentes notions relatives à la logique des prédicats d'ordre 0 et 1. Ce cours présente les méthodes de déduction sémantique et de résolution de problèmes. La deuxième partie concerne la gestion de bases de données relationnelles et l'utilisation du langage SQL. Dans ce cours, les étudiants apprennent à créer, consulter et interroger une base de données en utilisant les opérateurs de l'algèbre relationnelle (projection, restriction, intersection, etc).
  - Rôles : intervention TD, correction des examens.

2020-2021 **Enseignante-chercheuse LRU à l'université de Rennes 1**.

- **Introduction aux systèmes informatiques**
  - Public : L1 IE/MA
  - Volume horaire : 40h
  - Responsables de l'unité d'enseignement : Pierre Alain Fouque (pierre-alain.fouque@univ-rennes1.fr) et Patrick Derbez (patrick.derbez@univ-rennes1.fr)
  - Description : ce cours a comme objectif pédagogique d'apprendre à maîtriser les fondements de la programmation impérative (variables, types, boucles, fonctions, ...). Les notions de tests unitaire (JUnit) et de complexité sont légèrement survolées. À terme de ce module, les étudiants devraient être capables d'écrire des algorithmes et de les implémenter avec le langage Java.
  - Rôles : intervention TD et TP, participation à la préparation de supports pédagogiques, surveillance et correction d'examen, correction de rendus de TP de de projets.

- **Mise à niveau en algorithmique et programmation**
  - Public : L3 Info/Miage
  - Volume horaire : 12h
  - Responsable de l'unité d'enseignement : Mickaël Foursov (mikhail.foursov@irisa.fr)
  - Description : ce cours constitue une introduction accélérée aux bases de la programmation et au langage Java. Ces enseignements sont proposées aux étudiants qui viennent de parcours différents, aux étudiants étrangers et à tous les étudiants qui auraient besoin d'une mise à niveau en algorithmique et en programmation Java.
  - Rôles : intervention TP.

- **Programmation Objet**
  - Public : L2 Info/Miage
  - Volume horaire : 40h
  - Responsables de l'unité d'enseignement : Alexandre Termier (alexandre.termier@irisa.fr)
  - Description : ce cours s'adresse aux étudiants de deuxième année des licences Informatique et Miage qui sont supposés connaître les bases de la programmation. Ce cours introduit la programmation orientée objet (encapsulation, abstration, héritage, polymorphisme) en l'illustrant en langage Java. Par ailleurs, ce cours présente différentes structures de données (les listes chaînées et les arbres binaires de recherche) implémentées avec et sans les collections Java.
  - Rôles : intervention TP, évaluation orale des étudiants, surveillance d'examen en distanciel, correction de rendus de TP et projets.

- **Programmation immutable et fonctionnelle**
  - Public : L1 Info/Miage
  - Volume horaire : 82h
  - Responsable de l'unité d'enseignement : Delphine Demange (delphine.demange@irisa.fr)
  - Description : ce cours introduit
  - Rôles : intervention TD et TP, correction de rendus de TP, projets et contrôles.

- **Méthodes algorithmiques**
  - Public : L3
  - Volume horaire : 22h
  - Responsable de l'unité d'enseignement : Sophie Pinchinat (sophie.pinchinat@irisa.fr)
  - Description : ce cours se focalise sur les différentes méthodes algorithmiques existantes : diviser pour régner, les approches gloutonnes, programmation dynamique et linéaire. Pour chacune des méthodes, plusieurs exemples de problèmes sont présentés afin de

permettre à l'étudiant de comprendre la différence entre les approches. À terme, l'étudiant doit pouvoir proposer la solution adéquate pour résoudre différents problèmes.
  - Rôles : intervention TD.

- **Initiation au génie logiciel**
  - Public : L2
  - Volume horaire : 40h
  - Responsable de l'unité d'enseignement : Thomas Genet (thomas.genet@irisa.fr)
  - Description : ce cours est une initiation au génie logiciel proposé pour les étudiants en deuxième année informatique. Les notions principales de ce cours sont la conception logicielle, le développement à plusieurs en utilisant l'outil de développement collaboratif (Git), la génération de documentation et de tests unitaires.
  - Rôles : intervention TP, correction de rendus de TP, évaluation orale des étudiants.

2. Préparation de supports pédagogiques

Durant ma thèse, j'ai pu préparer quelques supports pédagogiques avec l'assistance de mon directeur de thèse Salah Sadou. Les supports étaient principalement pratiques destinés à des étudiants ingénieurs en sécurité (ENSIBS et Liban). Par ailleurs, j'ai l'opportunité de préparer un cours complet en sécurité logicielle à la **CyberSchool** de Rennes. Ce cours , qui commence la semaine 4 de 2021, est destiné aux étudiants de 2ème année en master international en sécurité. Ces étudiants ont déjà suivi un cours sur les vulnérabilités logicielles et ont des connaissances approfondies dans le domaine. Le cours que je propose, intitulé "Détection automatique des vulnérabilités logicielles", constitue une continuité du cours vu au premier semestre. Le contenu du cours s'articule en trois grandes parties : analyse statique, analyse dynamique et utilisation de techniques IA pour la détection de vulnérabilités logicielles. Ce cours comporte une partie théorique et une partie pratique.

3. Projet d'enseignement

Mon expérience d'enseignement et de recherche me conduit à enseigner différentes matières en informatique telles que l'algorithmique, la programmation, le génie logiciel et la gestion de bases de données. Par ailleurs, je suis fortement intéressée par des thématiques plus spécialisées telles que la sécurité informatique, et particulièrement la sécurité liée aux systèmes logicielles. Par ailleurs, ayant un master en systèmes informatiques intelligents, je suis vivement intéressée par l'enseignement dans ce domaine. Les matières qui m'intéressent le plus sont relatives à l'apprentissage automatique, les systèmes experts, la gestion d'agents intelligents et le traitement automatique de langage naturel.
Par ailleurs, afin de mener à bien mes enseignements, j'ai suivi plusieurs formations pédagogiques telles que "les questions de l'enseignant débutant", "comment préparer son cours ?" et "animer un cours interactif". Ces formations se sont avérées très utiles, d'autant plus dans le cadre de la crise sanitaire. Je suis fortement intéressée par les méthodes qui peuvent

être utilisées pour animer efficacement un cours en distanciel. À cet effet, j'organise des réunions régulières avec d'autres enseignants afin de partager mon expérience et expérimenter de nouvelles techniques proposées par mes collègues. Afin d'améliorer la qualité de l'enseignement numérique, j'ai pu mettre en place plusieurs solutions via l'utilisation de plateformes d'échange telles que *Moodle*, *Teams*, *Zoom*, des outils de partage tels que *Tableau noir*, *Codimd* et *OneNote*, l'utilisation de matériel dédié, notamment avec l'utilisation d'une tablette graphique. Je reste toujours ouverte à de nouvelles méthodes pouvant améliorer mon enseignement et rendre l'apprentissage plus agréable pour mes étudiants.

Service à la communauté et animation scientifique

En plus des conférences scientifiques où j'ai publié mes travaux, j'ai eu l'opportunité de participer à de nombreux évènements scientifiques :

- Participation à une journée de travail GLE/LOUISE/RIMEL organisée par le GDR GPL en 2016 (Nantes).
- Participation à la conférence nationale CIEL en 2017 (Montpellier).
- Participation aux journées de travail GLE/LOUISE/RIMEL en 2017 et 2019 (sans présentation).

En plus des événements scientifiques, j'ai participé dans des événements pour sensibiliser les jeunes femmes et les informer sur les métiers de l'informatique :

- Animation d'un atelier lors de la journée "Le numérique : des métiers en tous genres", IUT, Vannes (2019 et 2020)
  Cet événement était destiné à des jeunes collégiennes et avait pour objectif de leur présenter les choix possibles en informatique. Ceci s'inscrit dans un plan d'action pour la féminisation du secteur technique et informatique. Les collégiennes étaient très intéressées par les différentes activités et nous posaient des questions sur nos différents parcours.
- Talk lors de la journée "La femme et le numérique" , UBS, Lorient (2019)
  Cet événement n'était pas destiné aux femmes seulement mais portait également sur la problématique de la féminisation des recrutements. Des intervenant(e)s de plusieurs métiers, dits masculins, ont pu apporter leurs témoignages afin d'attirer les femmes à ces secteurs de travail.

Ces deux événements m'ont beaucoup marquées et inspirées. Je souhaiterais continuer sur cette lancée et mettre en place des événements similaires où des jeunes collégiens et lycéens peuvent découvrir les métiers de l'informatique. À mon humble avis, ces événements devraient être ouverts à toutes les personnes qui seraient intéressées par le métier de l'informatique et non destinés aux femmes seulement. L'idée serait de réduire le fossé existant et non pas de créer de nouveaux.

Liste de publications

Article long, 2019 **Designing a Code Vulnerability Meta-Scanner**.

ISPEC*19* - International Conference on Information Security Practice and Experience

- **Raounak Benabidallah**, Salah Sadou, Brendan Le Trionnaire, Isabelle Borne

Article long, 2019 **Designing a Code Vulnerability Meta-Scanner**.

CSET*19* - International Conference on Cyber Security for Emerging Technologies

- **Raounak Benabidallah**, Salah Sadou, Isabelle Borne

Article, 2017 **Identification de vulnérabilités dans les applications Java** .

Journées du GDR GPL, Actes de la Conférence en Ingénierie du Logiciel (CIEL)

- **Raounak Benabidallah**, Salah Sadou et Isabelle Borne

En cours de soumission **Using Software Quality Metrics to Predict Vulnerable Code**.

Journal of Systems and Software

- **Raounak Benabidallah**, Salah Sadou, Isabelle Borne et Gildas Menier