

## FORMATIONS **Docteur ès sciences (PhD)**

2020–2025

*École Polytechnique Fédérale de Lausanne (EPFL) - Suisse*

- Directeur de thèse: Prof. Mathias Payer au sein du laboratoire HexHive.
- Thèse: Sécuriser le code bas niveau avec un minimum d'efforts pour les développeurs.
- Thèmes: Sécurité des systèmes, tests logiciels, protections par les compilateurs, fuzzing.

## **Master of Science ETH en informatique**

2016–2019

*Eidgenössische Technische Hochschule Zürich (ETHZ) - Suisse*

- Spécialisation en Sécurité informatique, Moyenne: 5.39/6.

## **Bachelor en Systèmes de Communication**

2013–2016

*École Polytechnique Fédérale de Lausanne (EPFL) - Suisse*

- Année d'échange @ **Carnegie Mellon University** - USA, Moyenne: 5.26/6. 2015–2016

## **Maturité bilingue (Allemand/Français)**

2010–2013

*Kantonschule Frauenfeld & Gymnase d'Yverdon - Suisse*

- Option spécifique: Physique et application des mathématiques, Moyenne: 5.19/6, top 3%.

## EXPERIENCE **type++: prohibiting type confusion with inline type information**

NDSS'25

EN RECHERCHE *Auteurs: Nicolas Badoux, Flavio Toffalini, Yuseok Jeon, & Mathias Payer.*

- *Distinction des meilleurs articles* (top 5%).
- En C++, un downcast incorrect peut mener à des vulnérabilités sévères.
- En ajoutant un type à chaque objet C++, notre compilateur permet de vérifier chaque conversion. Prévenant tout risque de confusion de type à un nombre minime d'adaptations du code source. Nous obtenons moins de 1% de ralentissement tout en protégeant 90 milliards de conversions. Nous déployons notre prototype sur Chromium. Bâti sur LLVM, **type++** est disponible sur [GitHub](#) et son artefact a été évalué.
- En tant que leader de ce projet long de plusieurs années, j'ai acquis des compétences techniques, rédactionnelles, et stratégiques, par exemple, sur l'articulation d'un projet dans un domaine en constante évolution.

## **LIBERATOR: Balancing library fuzzing without consumer code**

FSE'25

*Auteurs: Flavio Toffalini, Nicolas Badoux, Zurab Tsinadze, & Mathias Payer.*

- Écrire des fuzz drivers, des séquences d'appel à une librairie pour du fuzzing, est complexe.
- LIBERATOR automatise leur création sans le besoin de code externe à la librairie et équilibre les ressources entre la création et le test des drivers. Via des passes LLVM, nous comprenons l'utilisation de la librairie et construisons des drivers C valides. Nous reportons 24 bugs, dont la CVE-2024-8006. Notre prototype est sur [Github](#).
- Pour l'évaluation multifacette ainsi que le design de LIBERATOR, j'ai dû anticiper les complexités futures et comprendre de manière transversale les caractéristiques des systèmes.

## **Sourcerer: channeling the void**

DIMVA'25

*Auteurs: Nicolas Badoux, Flavio Toffalini, & Mathias Payer.*

- En C++, les conversions entre **void\*** et des pointeurs typés sont courantes mais, si le type de destination diffère de celui d'origine, elles peuvent mener à de la mémoire corrompue.
- En étendant la protection de **type++** à tous les types, Sourcerer est le premier sanitizer complet pour ces erreurs. Avec un ralentissement de seulement 5% en moyenne, nous conduisons la première campagne de fuzzing visant spécifiquement les confusions de types.
- Sourcerer est disponible sur [GitHub](#) et trouve des erreurs dans Blender et OpenCV.
- Comme auteur principal, j'ai conçu l'architecture de Sourcerer, pris en charge l'évaluation et l'écriture de l'article.

## Bypassing LLVM-CFI cast protection

En cours

Auteurs: Nicolas Almerge, **Nicolas Badoux**, & Mathias Payer.

- Notre nouvelle attaque permet de contourner les protections de conversions de LLVM-CFI.
- Comme superviseur principal de ce projet de Master, j'ai défini le plan de recherche, guidé le travail, quantifié les résultats, et aidé à la rédaction du rapport.

---

EXPERIENCE **Ingénieur Informatique** - Fondation Digger, ONG - Tavannes, CH    *Août '19–Mars '20*

INDUSTRIELLE - Lors de mon service civil, développement, au sein d'un environnement Agile, d'une surcouche visuelle pour pouvoir détoner des mines avec une pelleteuse télécommandée.

**Ingénieur Informatique** - Compassion Suisse, ONG - Yverdon, CH    *Mars–Mai '18*

- Comme civiliste, j'ai contribué aux modules Python open source pour l'ERP Odoo.

**Ingénieur Informatique Stagiaire** - Ergon - Zürich, CH    *60%—Sept' '17–Mars '18*

- Création d'un fuzzer blackbox en Python pour tester le Web Application Firewall.

**Analyste Technologique Stagiaire** - Morgan Stanley - London, UK    *Juin–Août '16*

- Développement de tableaux statistiques en AngularJS pour l'équipe Sécurité.

---

COMPÉTENCES **Langages de programmation:** Python, C++,  $\text{\LaTeX}$ , Bash.

**Logiciels:** LLVM, Docker, GDB, Linux, libfuzzer.

**Langues:** Français (maternelle), Anglais, Suisse-Allemand, Allemand.

---

ASSISTANAT **CS-119 Information, calcul & communication**    *'22 & '24*

**CS-323 Systèmes d'opération**    *'21*

**CS-412 Sécurité logicielle**    *'21 & '23*

**COM-402 Sécurité de l'information & vie privée**    *'23*

---

ACTIVITÉS **Membre du Conseil, Trésorier** - Groupes Bibliques des Écoles et Universités *'23–présent*

- Définition de la stratégie, participation au processus de recrutement, et planification budgétaire ( $\simeq 500\text{kCHF}$ ).

**Directeur de Camp** - Interjeunes & Ligue pour la Lecture de la Bible    *'14, '17, '21, '22*

- Direction de plusieurs camps d'une semaine avec jusqu'à 110 enfants/jeunes adultes. Recrutement d'une équipe, préparation de l'événement, management de l'équipe, et en charge de l'autorité.

---

RÉFÉRENCES **Prof. Dr. Mathias Payer**    *mathias.payer@nebelwelt.net*

- Professeur Associé à l'EPFL à Lausanne (CH) et chef du laboratoire HexHive.
- Superviseur durant mon doctorat entre 2020 et 2025.

**Prof. Dr. Flavio Toffalini**    *flavio.toffalini@rub.de*

- Professeur Assistant à la Ruhr-Universität Bochum (DE).
- Proche collaborateur et post-doc durant la majorité de mon doctorat (2021–2025).

**Benoît Pfister**    *benoit.pfister@gbeu.ch*

- Président du Conseil des Groupes Bibliques des Écoles et Universités.
- Nous avons travaillé ensemble dans des comités d'engagement, pour le budget et la stratégie générale.