# Adrien **Ghosn**

PhD student in Computer Science

ghosn.adrien@gmail.com | https://people.epfl.ch/adrien.ghosn | aghosn | aghosn

## Education

### Ecole Polytechnique Federale de Lausanne(EPFL)

*Lausanne, Switzerland*

Computer Science Engineering — *Sep. 2010 - 2021*

- 2016 – 2021: PhD in Datacenter System Laboratory, Prof. Edouard Bugnion and Prof. James Larus
- 2013 – 2016: Master Degree, Foundations of Software specialization (avg 5.75/6)
- 2010 – 2013: Bachelor Degree

### Northeastern University(NEU)

*Boston, U.S.A.*

Master Thesis — *Sep. 2015 - Mar. 2016*

- Supervised by Prof. Jan Vitek in the Programming Languages Laboratory

### Carnegie Mellon University(CMU)

*Pittsburgh, U.S.A.*

Exchange Year, Bachelor Degree in Computer Science — *Aug. 2012 - Jul. 2013*

- Dean's list School of Computer Science for QPA > 3.75/4

## Industry

### Researcher

*Cambridge UK*

Microsoft Research — *November 2021 - present*

- Verona software- & hardware-based sandboxed libraries
- Verona operational semantics interpreter
- Tyche: unified software TEE on legacy hardware

### Summer Internship

*Kirkland, USA*

Google Asylo team - Matt Gingell — *June - August 2019*

- Asylo team, Trusted Execution environments, SGX
- Explored potential designs to support higher-level programming languages in SGX enclaves
- Delivered a prototype that allowed HLPL code to run inside SGX

## Research & Publications

**Focus Areas**

Programming Languages, Systems, Security
Isolation of mutually distrustful software components
Hardware-enforced isolation

### Ongoing Research

*Cambridge, UK*

MSR: Marios Kogias, EPFL: Prof. Edouard Bugnion, Prof. James Larus, Prof. Mathias Payer — *Nov. 2021 - Present*

- Tyche: supporting all TEE programming models on legacy hardware

### Enclosures: Language-based restriction of untrusted libraries [1]

*Lausanne, Switzerland*

EPFL, DSCL - Prof. Edouard Bugnion, Prof. James Larus, Prof. Mathias Payer — *Sep. 2019 - Oct. 2020*

- New fine-grain programming abstraction to restrict public libraries access to program resources
- Frontend extensions to Go and Python PLs
- Backend support for hardware isolation enforcement (Intel VT-x & Intel MPK)
- Intra-address-space isolation, Sandboxing, Compiler, Linker, Runtime

### Secured Routines: Language-based construction of TEEs [2]

*Lausanne, Switzerland*

EPFL, DSCL - Prof. Edouard Bugnion, Prof. James Larus — *Jun. 2018 - May 2019*

- Extended Go programming language to support executing goroutines inside Intel SGX.
- Intel SGX, Confidentiality, Intergrity, Go, Compilers, Code partitioning, Hardware Extensions

### Light-Weight Contexts in Dune

*Lausanne, Switzerland*

EPFL, DSCL - Prof. Edouard Bugnion — *Sep. 2016 - Jul. 2017*

- Process virtualization with Dune
- Intra-address space isolation, protecting secrets, memory snapshots
- 5x speed improvement over a Linux fork
- Intel VTX, Dune, Virtualization, Kernel module, Virtual Memory Management

### Efficient Runtime Deoptimization for R(Master Thesis)

*Boston, U.S.A.*

Northeastern University - Prof. Jan Vitek — *Sep. 2015 - Mar. 2016*

- Speculative optimizer for an R JIT compiler
- Removes performance bottlenecks due to the language semantics
- Ensures correct run-time behavior.
- On-stack replacement, speculative optimizations, runtime de-optimization, R, LLVM, JIT compiler

### Aperiodic-Event Support in FASA

*Baden, Switzerland*

ABB Corporate Research - Dr. Manuel Oriol — *Feb. 2015 - Aug. 2015*

- Fixed-priority servers, data-driven events, real-time control applications
- kernel design, dynamic linking/loading & software updates, pi-calculus

### Scalameta: AST Persistence & Obey: Code Health

*Lausanne, Switzerland*

EPFL, LAMP - Prof. Martin Odersky & Dr. Eugene Burmako — *Jan. 2014 - Feb. 2015*

- Obey: Scala-linter for user-defined rules enforced at compile-time
- AST Persistence: typed-AST format for Scala
- Resolves compiler version incompatibilities and provides IDE macros expansion support

## Projects

### Operating Systems & Design 15-410

*CMU*

Undergraduate — *Jan. 2013 - Jul. 2013*

- Implementation of a x86 Unix like Kernel in C and ASM
- Design and implementation of thread library, scheduler, virtual memory, various drivers, system calls

### Tweet Aggregator

*EPFL*

Graduate — *Jan. 2014 - Jul. 2014*

- Big Data web application to gather and display real-time tweets on a map, according to user-defined keywords
- Filtering and clustering tweets according to zoom-level and selected geographical areas

### Compiler & Advanced Compiler

*EPFL*

Graduate — *Sep. 2013 - Jul. 2014*

- Design & implementation of compilers for JVM-based Lisp-like languages
- Mark & sweep garbage collector and optimization phases including DCE-CSE, constant folding, closure hoisting

## Skills

| | |
|---|---|
| **Programming** | Go, C/C++, Java, Shell scripting, asm, Python |
| **Knowledge in** | Compilers & PL design, Language runtimes & virtual machines<br>Operating System design, Virtualization, KVM, Intel VT-x, Intel MPK<br>Software security, Hardware Security extensions, Trusted Execution Environments<br>Theoretical CS, Concurrent & Distributed Algorithms |

## Teaching

| | |
|---|---|
| **Semester Projects** | Go Intel MPK library (Charly Castes)<br>System call filtering in Go & Python enclosure support (Elsa weber) |
| **Teaching Assistant** | Functional Programming (2020), Introduction to Operating Systems (2019)<br>Introduction to Java Programming (2018), Systems for Data Science (2017-2020)<br>Introduction to C Programming (2016-2017), Concurrent Programming (2015)<br>Student Volunteer at ECOOP (2016) |

## Personnal

**Languages** Fluent in French & English

## References


[1] Adrien Ghosn, Marios Kogias, Mathias Payer, James R. Larus, and Edouard Bugnion. Enclosure: language-based restriction of untrusted libraries. pages 255–267, 2021.

[2] Adrien Ghosn, James R. Larus, and Edouard Bugnion. Secured Routines: Language-based Construction of Trusted Execution Environments. pages 571–586, 2019.