Formally Verified Security @ MPI-SP

1. Security Goal

2. Enforcement

3. Formal Validation

- **Cătălin Hriţcu** (Tenured Faculty)
- **Cezar Andrici** (PhD student)
- **Jonathan Baumann** (PhD student)
- **Yonghyun Kim** (PostDoc)
- **Julay Leatherman-Brooks** (Intern)
- **Abigail Pribisova** (CS@max planck)
- **Jérémy Thibault** (visitor, graduated PhD)

Former group members now faculty or permanent researchers:
Roberto Blanco (Assoc. Prof., TU/e), **Lef Ioannidis** (Senior RSE, MSR), **Guido Martínez** (RSE, MSR), **Théo Winterhalter** (Tenured Researcher, Inria), **Carmine Abate** (Senior Researcher, Barkhausen), **Adrien Durier** (Assoc. Prof, Univ. Paris-Saclay), **Kenji Maillard** (Tenured Faculty, Inria), **Danel Ahman** (Assoc. Prof, Univ. of Tartu), **Arthur Azevedo de Amorim** (Assist. Prof, RIT), **Marco Stronati** (Research Scientist, Matter Labs), **Clément Pit-Claudel** (Assist. Prof, EPFL), **William Bowmann** (Assist. Prof, UBC), **Diane Gallois-Wong** (RSE, Nomadic Labs), **Zoe Paraskevopoulou** (Assist. Prof, NTU Athens), **Nick Giannarakis** (Senior Applied Scientist, AWS), **Thorsten Tarrach** (Senior Applied Scientist, AWS)

Secure compilation of verified F* code

1. **Very strong guarantee**, stronger than full abstraction
2. **Reference monitoring** and **higher-order contracts**
3. **Machine-checked proofs in F***

[Cezar et al, TYPES'22, HOPE'22, POPL'24, ICFP'25]

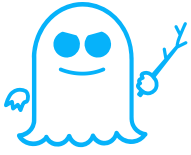
Other interesting topics on F*

- Dijkstra monads and incorrectness logic
- Dijkstra Monad for Bounding Failure Probability (crypto proofs)
- Separation logic in F* (Pulse)
- F* foundations: demystifying ghost and divergence effects

Secure compilation of compartmentalized C code

1. **Restricting scope of UB** to compromised compartments
2. **CompCert** variant to **CHERI RISC-V** capability machine
3. **Scalable machine-checked proofs in Rocq**

[Jérémy et al, CCS'18, CSF'19, ESOP'20, CSF'22, CCS'24, ITP'25]



Stronger Security Goals

Preserve data confidentiality

for compartmentalized programs in F*, C, Rust, or Wasm

Realistic Enforcement

ARM Morello capability machine

Capability passing

Better Proof Techniques

Verify capability backend

FS-CASA: Formally Secure Compilation Against Spectre Attacks

1. Relative security

– compiled program doesn't leak speculatively more than what (arbitrary!) source program leaks sequentially

2. Building on FSLH: Flexible Speculative Load Hardening [Jonathan et al, CSF'25]

– Extending this to all main Spectre variants

– Want to implement this defense in LLVM

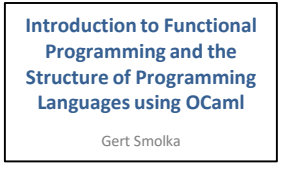
3. Testing and proving relative security

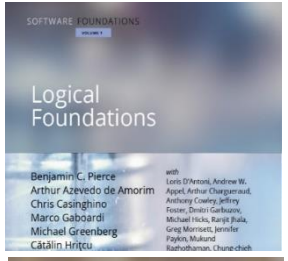
– Building new Property-Based Testing framework for LLVM and x86 (HW/SW contracts)

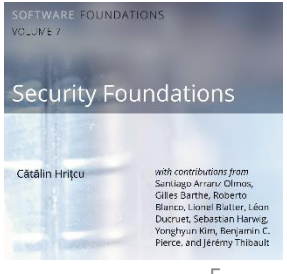
– Constructing machine-checked proofs in Rocq for simplified models

Courses we teach in Bochum and Remote

1. **Functional Programming** (Winter 2025/26)
2. **Proofs are Programs** (Summer 2026)
3. **Foundations of Programming Languages, Verification, and Security** (Winter 2026/27?)







Clara Schneidewind, Cătălin Hriţcu, Jana Hofmann

Max Planck Institute for Security and Privacy (MPI-SP)