Formally Verified Security

Cătălin Hriţcu

MPI for Security and Privacy
in Bochum

Formally Verification is Getting Real

Firefox

Mozilla shipping EverCrypt verified crypto library

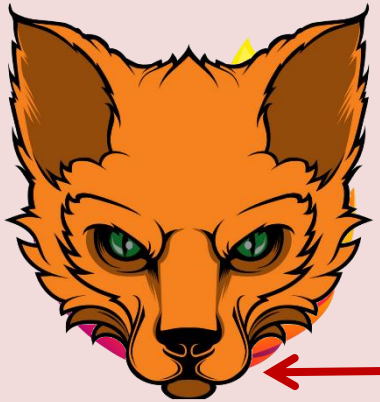
(also used by Microsoft, Linux, ...)

Formal verification milestone:

40.000+ lines of highly-efficient code, proved to be free of vulnerabilities, functionally correct, and side-channel resistant

Formal Verification is Still Limited

Firefox



Can we protect?

EverCrypt

40.000 lines of F*

20.000.000 lines
unverified and vulnerable

insecure interoperability:

if Firefox is compromised it can break security of verified code

Formally Verified Security

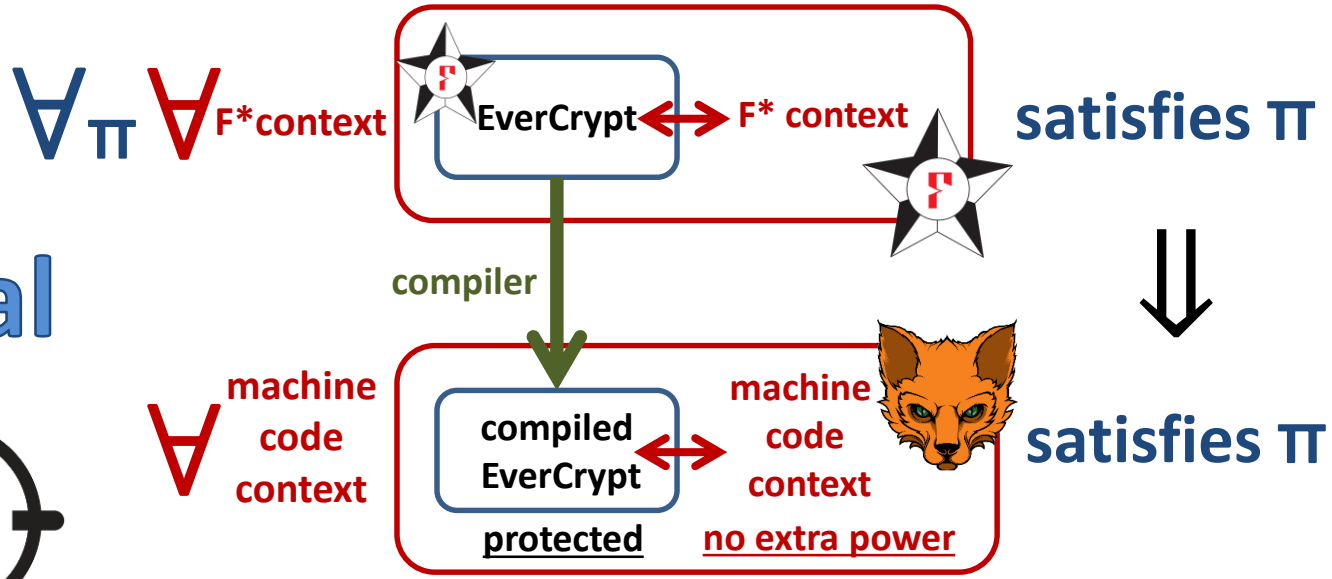
Proof

Goal

Enforcement

1. Security Goal

Formal security definition for secure compilation



2. Security Enforcement

many cool abstractions:
types, modules, functions,
effects, specifications, ...

Secure language

secure
compiler
chain

CHERI capability machine

inspiration for
ARM Morello

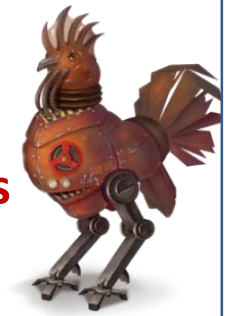
Hardware-accelerated enforcement

to achieve our goal need to
protect these abstractions
all the way to machine code

3. Security Proof

Formally verifying the security of such compilation chains

- such proofs **very difficult** (wrong conjec
  and tedious (e.g., 250 page
- **more scalable proof techniques**
- **develop proofs as programs**
  - **machine-checked proofs in the Coq/F* proof assistants**
- **recently applied to realistic compiler for compartmentalized C code** (based on CompCert)



Cătălin Hriţcu
MPI for Security and Privacy in Bochum