The CertiKOS Project

Zhong Shao
Yale University
April 21, 2017
http://flint.cs.yale.edu

Acknowledgement: Ronghui Gu, Newman Wu, Hao Chen, Jieung Kim, Jeremie Koenig, Vilhelm Sjoberg, Mengqi Liu, Lionel Rieg, Quentin Carbonneaux, Unsung Lee, Jiyong Shin, David Costanzo,
Tahina Ramananandro, Hernan Vanzetto, Shu-Chun Weng, Zefeng Zeng, Zhencao Zhang, Liang Gu,
Jan Hoffmann, Joshua Lockerman, and Bryan Ford. This research is supported in part by DARPA
CRASH and HACMS programs and NSF SaTC and Expeditions in Computing programs.
Hacker-Resistant OS: Why?

Crash

Mobile

OS

Applications

Hardware

Life

Loss

Environment

Financial

cloud
OS Landscape (Jan 2017)

Desktop: Linux, macOS, Windows, ChromeOS, FreeBSD, …
Hypervisor/Cloud: Linux KVM & Docker, VMWare, Xen, …
Mobile: Android (Linux), iOS, …
Embedded: Embedded Linux, VxWorks, QNX, LynxOS, …

- All of them are bloated, old, and contain many bugs
- Urgently need new OSes for emerging platforms & apps
  (IoTs, Drones, Self-Driving Cars, Cloud, NetworkOS, Blockchains, …)

OS evolution has reached an inflection point:
Need a certified “hacker-resistant” OS that provides security, extensibility, performance, and can work across multiple platforms.
Hacker-Resistant OS: How?

Problems w. existing platforms

- Attacks: Zero-Day Kernel Vulnerabilities (ZDKVs) & rogue driver certificates
  - leads to rogue kernels
  - leads to rogue apps & drivers
  - leads to rogue PLC firmware

New CertiKOS technologies

- A small certified “hypervisor” kernel provides a reliable ZDKV-free core to fall back on, even under attacks
  - Information-Flow-Control to enforce security
  - Mechanized proof certificates are unforgeable
mCertiKOS (2015)

Single-core version of CertiKOS (developed under DARPA CRASH & HACMS programs), 3 kloc, can boot Linux

Aggressive use of abstraction over deep specs (37 layers in ClightX & LAsm)
Concurrent CertiKOS [OSDI’16]

The first fully certified concurrent kernel w. fine-grained locking
The CertiKOS Project

**Killer-app:** high-assurance “cyber” systems (of systems)!

**Long-term goals:** Can we build a new certified OS and programming platform --- a certified version of Linux-like extensible kernel with a foundational layer providing a reliable and secure substrate for future computing?

**Short-term goals:**
- New certified system software stacks (CertiKOS +)
- New certifying programming languages (DeepSEA vs. C & Asm)
- New certified programming tools
- New certified modeling & arch. description languages
- We verify all interesting properties
Hacker-Resistant OS: Why Yale?

Very high barriers of entry:

1. OS kernel development is very difficult
2. Formal specifications and proofs are hard to build
3. Need intimate programming language expertise to succeed
   - These are three completely different communities
   - Most people can only do one out of the above three.
   - The PI’s team has been working on all three for >20 years

CertiKOS is the first fully certified OS kernel that is done economically (< 3 person years), proves more properties, runs on concurrent HW, and is truly extensible

- The main competition (seL4 in Data61): took 22 person years; not extensible; does not support concurrency; does not use Coq (thus lacking support from tools such as CompCert)