A New RISC
Demo Time
Who Am I

• DARPA grant winner for IoT
• GSMA IoT Guidelines author
• First car hacker
• Etc
What do I do Now?

- Secure IoT platforms
- IoT as DevOps
- Evolving the Internet
Privilege Model
Four Layers

- User (lowest privilege)
- Supervisor (OS kernel here)
- Hypervisor (deprecated)
- Machine (highest privilege)
Obvious Attack Model

- Transition from User -> Supervisor is a separate issue
- Supervisor -> Hypervisor or Machine = Goal
- Machine inevitable goal; control all system resources
Current Attack Model

- Find physical base address
- Machine resident executive lives in physical memory
- Supervisor lives in virtual memory
- Machine layer sets up initial page table directory
- But Supervisor controls page tables to manage user tasks directly
- This means Supervisor code can alter **all** virtual memory
Attack Implementation

- Gain execution privilege in the Supervisor layer
- Create a virtual memory map for Machine layer
- Overwrite the Machine layer's trap table
- Flush the TLB
- Trigger a trap to Machine layer
Making it Clean

- Can't just map arbitrary Machine executable code
- Must overwrite something useful
- Can't overwrite critical code (deadlock or system panic / reset)
- Choose a stupid or unused feature
- In my case the MCALL_SHUTDOWN system call
The Solution

- Restrict physical memory accessible from Supervisor layer
- Machine layer defines physical memory region limits
- Supervisor can't access Machine physical memory or registers
- Requires silicon change
- PMP proposal introduced by RISC-V team 15 days ago!
What is RISC-V
The Death of ARM?

- Open source ISA
- No per chip royalties! No up-front pay to play model!
- Freely available FPGA implementations
- Open development environment
- Very accessible silicon engineering
Tagged Memory
What is Tagging?

- LowRISC implementation
- Marks objects in memory with metadata
- Metadata tells silicon how to manage objects
- e.g. Return addresses can't be overwritten directly
Limitations

- Requires more silicon
- Results in higher cost
- Requires tool chain improvements (support tag instructions)
- Supervisor needs to handle tag traps
- Ecosystem changes to handle failures
Protecting Debugging
Traditional Attacks

- Bypassing fuses (glitching and such)
- Racing the firmware for control
- Guessing a debug password
- Extracting passwords from flash
Solution?

- Using the existing authentication model in RISC-V
- Two registers available before the core is live
- Spec define basic auth (bitlength "password")
- Augment to use public key authentication
- Silicon based solution for SoC with sufficient IP
- Allows serial communication over auth registers for extension
Application Attacks
Architecture Generics

- Pretty much the same as any RISC
- 16-bit opcodes require no context switch (like Thumb does)
- Embedded system definition is slightly different
- Otherwise quite easy
Thank you!

SiFive Team
Stefan O'Rear
Ron Minnich
Megan Wachs
https://github.com/donbmouse/riscv-security/

donb@securitymouse.com