HOST
Hardware Roles in Driving System Security

James Fahrny
Senior Fellow, Security Research
May 4, 2016

COMCAST

Trust and Intrusion Detection

- Intel has helped lead some advances with Trusted Execution (TXT) and Enclaves along with security in Sandy/Ivy Bridge, Haswell, Skylake!
- We don't have Intel processors in all solutions!
- Even the general purpose processor needs to have trusted execution/IDS
- What we really need:
  - **Hardware Root of Trust (Secure Processor, Secure OTP, Trusted/Private Key storage)**
  - **Cryptographic Control Flow Integrity**
  - **Hardware code signing or HMAC generation and validation**
  - **Symmetric unique keys in every device (Stable PUFs and/or key delivery solution)**
  - **Royalty Free or Royalty Friendly development of these components**

Critical Hardware – Moving Forward

One Time Programmable (OTP)

- Poly-fuse and e-Fuse allow Secret Key and Identity extraction/modification
- Kilopass or e-Memory technologies are required to obfuscate the root keys
- We really need to fund some additional research in this area for low cost devices and for the high end devices
- Root keys need to be used to derive a key and never exposed (secure ladder)
- Once Root keys are compromised, the device is dead unless there are security

methods employed to vote in a new root key key and revoked the previous root.

Cryptographic Control Flow Integrity(CCFI)

- Control Flow should not deviate from its control flow graph
- Anytime an address is written or copied to memory, compute and append 64 bit AES-MAC
- Before execution of address (Stack and Heap), verify MAC and fail/crash if failure

Critical Hardware – Moving Forward

Cryptographic Control Flow Integrity(CCFI) - cont.

- What about performance? 3% to 18% slowdown on non-cryptographic processors.
- How do we correct this performance: Heavy use of AES-NI!
  - 2013 Haswell: 7 cycles
  - 2015 Skylake: 4 cycles (fully pipelined)
  - 2017 Kaby Lake: 2 cycles
- Fast AES enables new unexpected applications!
- We need to have CCFI acceleration/security capability in other processors and hardware

COMCAST

Critical Hardware – Moving Forward

Cryptographic runtime Code Signing or HMAC

- We need to be able sign data sections and validate in background (on the fly)
- Integrity checking does not needs to happen before every use but should be flagged if compromised memory writes occur.
- Ideally if root file system, OS page swaps and device drivers are validated before each use.
- Applications/executable code would all required to be signed/HMAC with a set of Permissions and privileges that are stored in protected memory/hardware.
- **These privileges would include:**
  - **Drivers permitted access (disk write/read, Network Communication, etc)**
  - **Memory region access read and write privileges**
  - **Permissions for communication to other applications and access to kernel functions**
  - **Root File system read and write privileges**
  - **Spawning/forking processes**

COMCAST

Intrusion Detection or Hypervisor or….?

- We need to sign data sections and validate in background
- Integrity checking does not need to happen before every use but should be flagged if memory writes are compromised.
- I'm not a fan of hypervisors…since they run at higher level and less trusted
- There is a concept of "metavisors" that run inside or under the OS
- This security agent even has control the privileges for the Root User/Admin
- This agent could control whether a shell can be spawned, whether the kernel has been modified and even whether the root file system has been hacked.
- 300K VMs running in cloud...500K IP addresses...**Can you detect intrusion?**
- How do you securely communicate the attack and respond to the attacks?

Summary/Next Steps

- Drive containers like Dockers or Rocket into all enterprise software
- Funding Research in these areas: OTP, CCFI, HW Code Signing
- Develop real-time IDS or metavisors solutions for systems/hardware.
- We need to define the process for symmetric key distribution

and/or make PUFs stable over time and get error rates approaching 0!

- Specifications and research is needed in these areas!

- Royalty Free or Royalty Friendly

Questions or Comments?

COMCAST