

AMD's Virtualization Memory Encryption ▲

KVM Forum 2016
August 25, 2016

AGENDA



- ▲ Technology
- ▲ Key Management
- ▲ Integration

Technology ▲

- ▲ **Hypervisor must enforce full isolation between co-resident VMs**
 - Typically using hardware virtualization support like AMD-V™ Technology
 - “Logical isolation” using page tables, VM intercepts, etc.
 - Sometimes breaks down
 - QEMU “VENOM” (CVE-2015-3456)
 - VirtualBox bug (CVE-2014-0983)
 - Etc.

- ▲ **Cloud users must fully trust the cloud hosting company**
 - Hypervisor has full access to guest secrets in memory
 - Hypervisor enforces all isolation
 - Not ideal for users *or* cloud companies

HARDWARE MEMORY ENCRYPTION - ATTACKS



DEFENDED BY AMD SECURE MEMORY ENCRYPTION + AMD SECURE ENCRYPTED VIRTUALIZATION

User Access Attacks

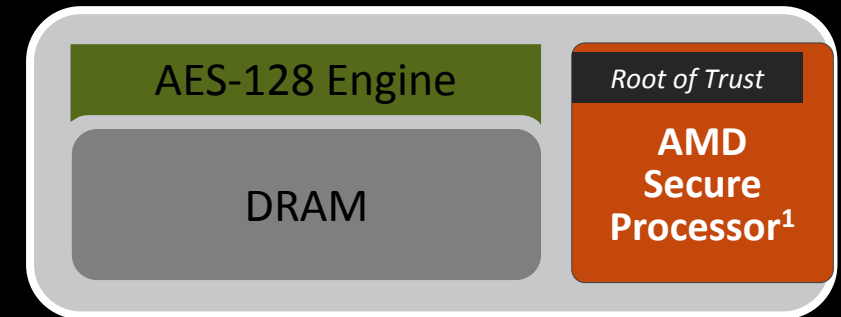
- Administrator scrapes memory of guest data areas
- Administrator injects code into a guest VM
- Hypervisor bug allows hosted guest to steal data from other guests

Physical Access Attacks

- Probe the physical DRAM interface
- Install HW device that accesses guest memory
- Freeze then steal DIMMs
- Steal NVDIMMs

AMD Secure Memory Encryption / AMD Secure Encrypted Virtualization

- ▲ Hardware AES engine located in the memory controller performs inline encryption/decryption of DRAM
- ▲ Minimal performance impact
 - Extra latency only taken for encrypted pages
- ▲ No application changes required
- ▲ Encryption keys are managed by the AMD Secure Processor and are hardware isolated
 - not known to any software on the CPU



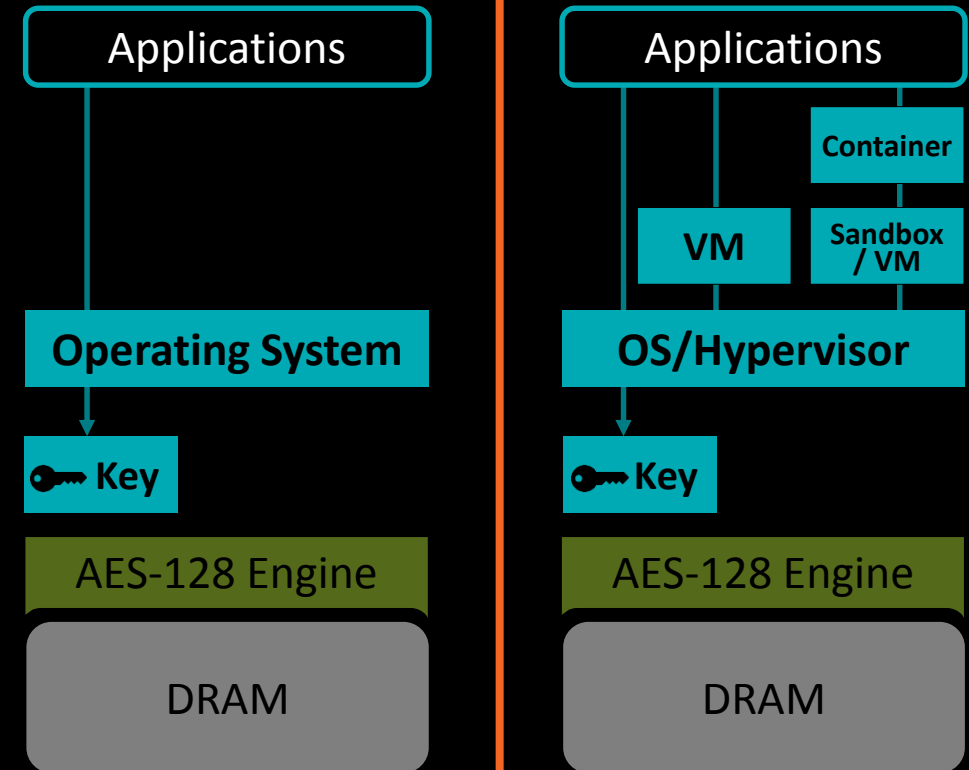
Defense against unauthorized access to memory

HW MEMORY ENCRYPTION – AMD SECURE MEMORY ENCRYPTION



- ▲ Helps protect against physical memory attacks
- ▲ Single key is used for encryption of system memory
 - Can be used on systems with VMs or Containers
- ▲ OS/Hypervisor chooses pages to encrypt via page tables
- ▲ Support for hardware devices (network, storage, graphics cards) to access encrypted pages seamlessly through DMA

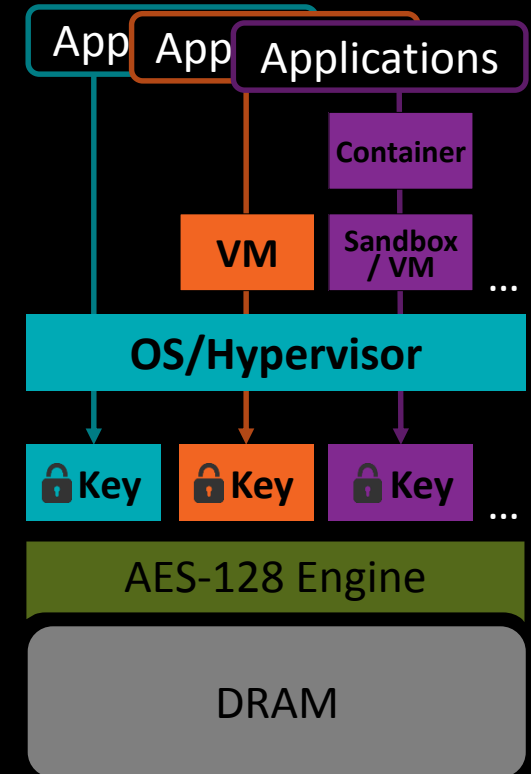
Added defense against unauthorized access to memory



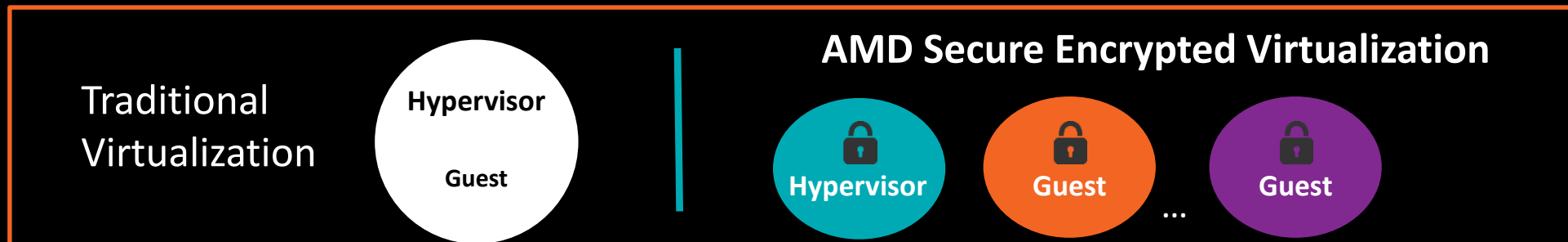
HW MEMORY ENCRYPTION – AMD SECURE ENCRYPTED VIRTUALIZATION

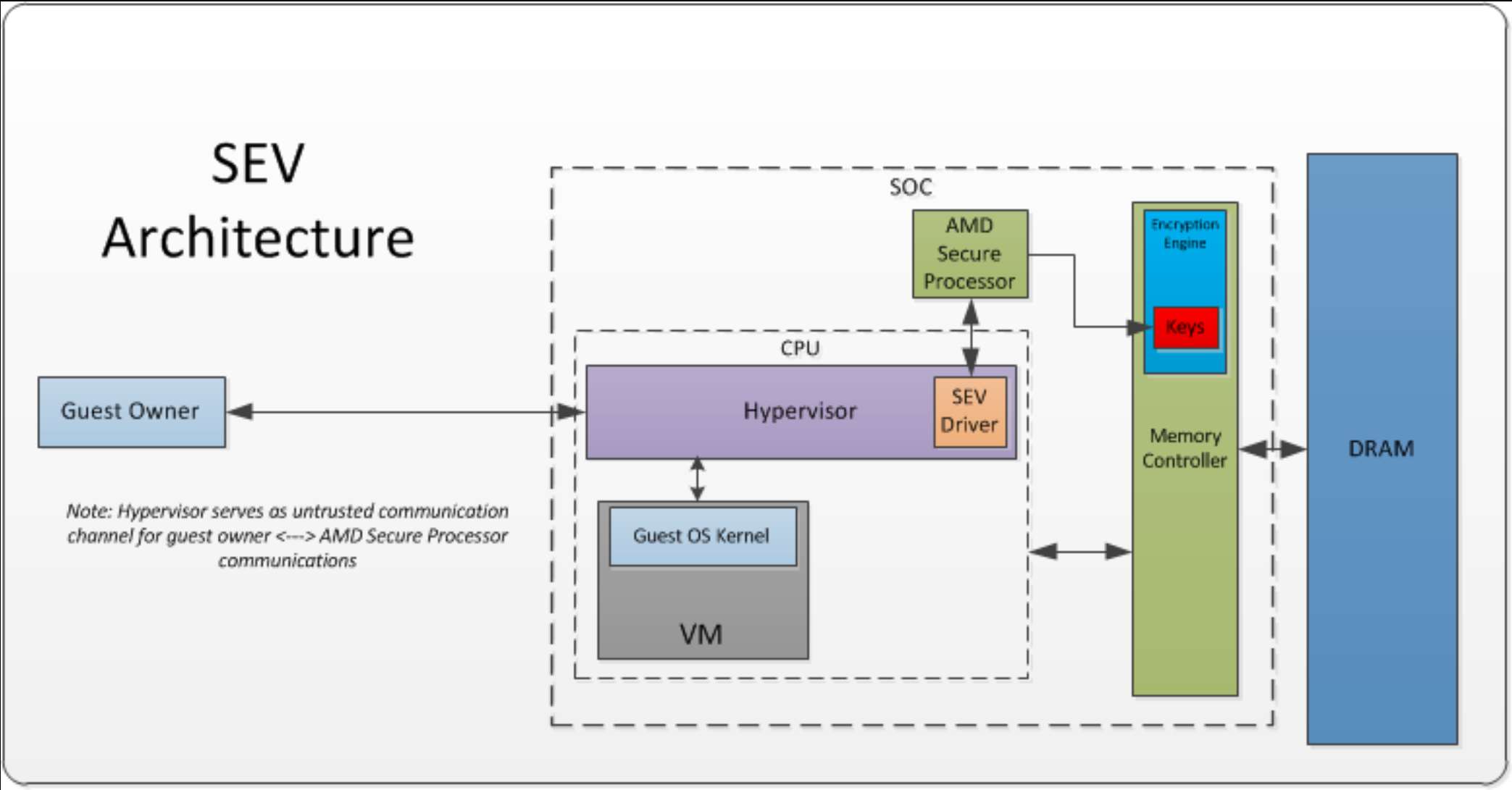


- ▲ Protects VMs/Containers from each other, administrator tampering, and untrusted Hypervisor
- ▲ One key for Hypervisor and one key per VM, groups of VMs, or VM/Sandbox with multiple containers
- ▲ Cryptographically isolates the hypervisor from the guest VMs
- ▲ Integrates with existing AMD-V™ technology
- ▲ System can also run unsecure VMs



Enhances isolation of VMs





- ▲ Address Space ID (ASID) determines VM encryption key
 - ASID is tagged with all data within the SoC
 - ASID determines encryption key to use when data enters/leaves SoC

- ▲ HW and Guest page tables determine if a page is “private” or “shared”
 - Instruction code pages always “private”
 - Guest page tables always “private”
 - Data pages can be “private” (C=1) or “shared” (C=0) depending on page tables
 - Before CR4.PAE=1, all pages are “private”

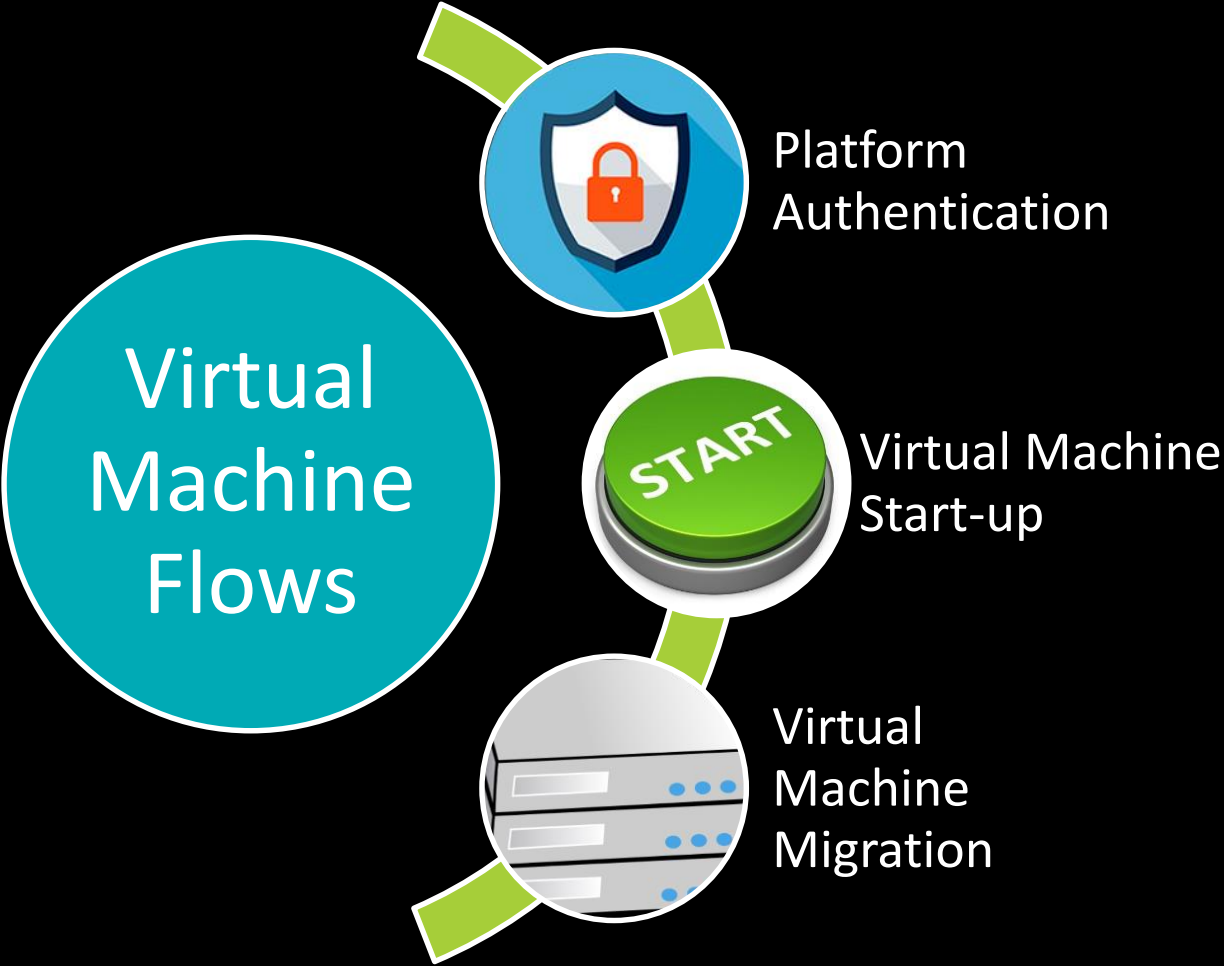
- ▲ All DMA must occur to “shared” pages

- ▲ Example use: all guest pages are “private” except for DMA pages

Key Management ▲

SEV KEY MANAGEMENT

VM LIFECYCLE INTEGRATION

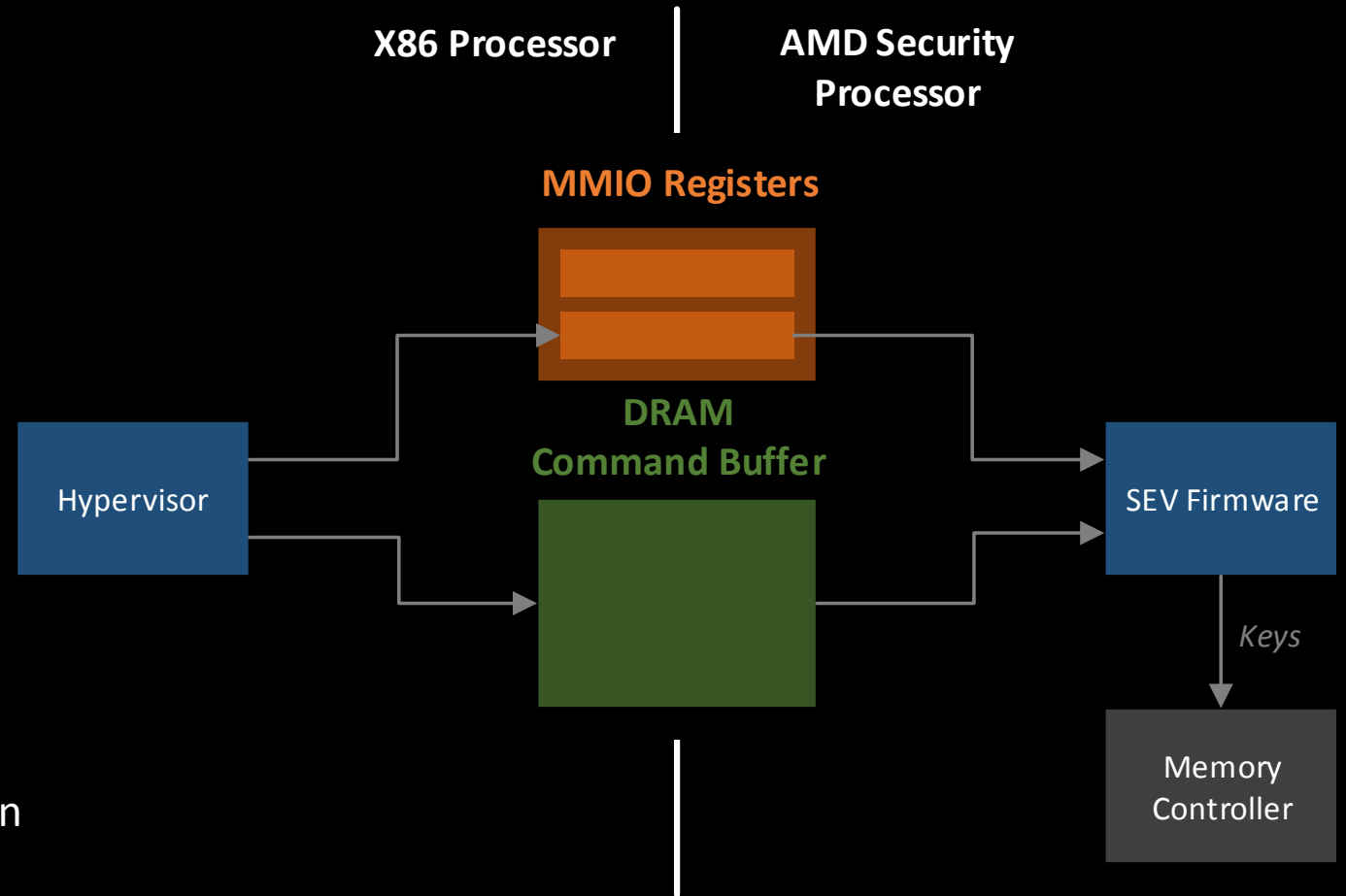


SEV KEY MANAGEMENT

ARCHITECTURE



- ▲ Firmware executes on the AMD Secure Processor
 - Isolated from x86 software
- ▲ Communicates with x86 software
 - Mailbox registers
 - Shared memory buffers
- ▲ Assists hypervisor in VM lifecycle
 - Generates and manages encryption keys
 - Bootstraps memory encryption during guest launch
 - Prepares guest memory image for transmission before migration (or snapshot)
 - Receives guest memory image after migration (or snapshot)
- ▲ Enforces guest policy



SEV KEY MANAGEMENT

GUEST LIFECYCLE



- ▲ Platform Key/Certificate Management
 - Authenticity and ownership of the platform
- ▲ Launching
 - Guest images created with unencrypted components
 - Need to bootstrap encryption before enabling SEV
- ▲ Migration and Snapshot
 - Support typical migration and snapshot operations
 - Protect guest memory image during transmission and storage
 - Prevent sending guest to an untrusted platform
- ▲ Activation
 - Associate an ASID with a guest and its memory encryption keys
 - Allows overcommitting of key slots

Integration ▲

▲ Key Management API

- New IOCTL support in KVM to launch guests, migrate guests, etc.
 - Unwrap and encrypt guests for execution
 - Wrap/unwrap guest memory pages for migration
 - Invoke AMD Secure Processor driver to perform communication with the AMD Secure Processor
- Updates to virtualization tools (libvirt, etc.)
 - Initialize platform
 - Store and provide guest key material
 - Return guest measurements

▲ Key Management API

▲ ASID Management

- SEV guests must have the same ASID for all vCPUs
 - Requires TLB flush if a different vCPU for the same ASID is to be run on the same host CPU
- SEV guests must have an ASID value within specified range
 - SEV ASID range obtained through CPUID instruction
- Non-SEV guests can use any ASID
 - Should use a value outside the SEV ASID range to avoid reducing available SEV resources

- ▲ Key Management API
- ▲ ASID Management
- ▲ Debug Support
 - Controlled through guest policy
 - Allows for QEMU to encrypt/decrypt guest memory
 - Maintains compatibility with current QEMU debug techniques

- ▲ Key Management API
- ▲ ASID Management
- ▲ Debug Support
- ▲ Paravirt Drivers
 - VirtIO
 - Requires “shared” pages for any memory that the HV needs to access
 - Virtqueues
 - Buffers used by HV to perform data operation
 - KVM Clock
 - “Shared” page in early boot
 - Others...

- ▲ Key Management API
- ▲ ASID Management
- ▲ Debug Support
- ▲ Paravirt Drivers
- ▲ DMA
 - Must be performed to “shared” pages
 - Make use of SWIOTLB to go between “private” and “shared” pages

▲ AMD is developing

- AMD Secure Processor firmware to implement key management tasks (distributed in AGESA)
 - Signed by AMD, source not public
- Linux driver to facilitate HV to AMD Secure Processor communication
 - Open source

▲ Other major components

- Linux kernel support for AMD Secure Memory Encryption and AMD Secure Encrypted Virtualization
 - RFC patches have been sent to LKML
- KVM/QEMU support
 - Managing ASIDs, facilitating guest owner communication, etc.

- ▲ Whitepapers (<http://developer.amd.com/resources/documentation-articles/articles-whitepapers/>)
 - [AMD Memory Encryption](#) – Overview of AMD Secure Memory Encryption and AMD Secure Encrypted Virtualization features

- ▲ Manuals/Specifications (<http://developer.amd.com/resources/documentation-articles/developer-guides-manuals/>)
 - [AMD64 Architecture Programmer's Manual Volume 2: System Programming](#) (sections 7.10 and 15.34)
 - [Secure Encrypted Virtualization Key Management](#)

Thank You!

DISCLAIMER & ATTRIBUTION



- ▶ 1. AMD Secure Processor (formerly “Platform Security Processor” or “PSP”) is a dedicated processor that features ARM TrustZone® technology, along with a software-based Trusted Execution Environment (TEE) designed to enable third-party trusted applications. AMD Secure Processor is a hardware-based technology which enables secure boot up from BIOS level into the TEE. Trusted third-party applications are able to leverage industry-standard APIs to take advantage of the TEE’s secure execution environment. Not all applications utilize the TEE’s security features. AMD Secure Processor is currently only available on select AMD A-Series and AMD E-Series APUs. GD-72

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions and typographical errors.

The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION.

AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY DIRECT, INDIRECT, SPECIAL OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

ATTRIBUTION

© 2016 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo and combinations thereof are trademarks of Advanced Micro Devices, Inc. in the United States and/or other jurisdictions. Other names are for informational purposes only and may be trademarks of their respective owners.