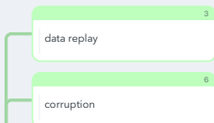
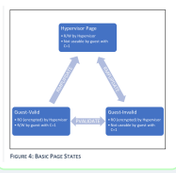
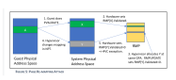
SEV-SNP-strengthening-vm-isolation-with-integrity-protection-and-more-1e2d0f1f63c9168a575bd487c8c8698b

- Secure Encrypted Virtualization (SEV)
  - minimize their exposure to bugs in the cloud provider's infrastructure
  - keep their data confidential
  - isolate VMs at a hardware level
    - individual VMs could be assigned a unique AES encryption key
- SEV-ES (Encrypted State)
  - protection for CPU register state
- SEV-SNP (Secure Nested Paging)
  - Threat Model
    - 
      - fully trusted
        - AMD Secure Processor (AMD-SP)
        - VM
      - fully untrusted
        - all other CPU software components
          - BIOS on the host system
          - hypervisor
          - device drivers
          - other VMs
        - PCI devices
    - SEV technologies only protect data in-use
    - 
    - single data structure shared across the system
    - one entry for every 4k page of DRAM that may be used by VMs
    - tracks the owner for each page of memory
    - in conjunction with standard x86 page tables
  - Reverse Map Table
    - New CPU instructions exist to enable manipulation of RMP entries
      - PVALIDATE
        - sets the Validated bit
      - RMPUPDATE
        - assigns the page to the guest
      - 
      - RMPADJUST
        - allows a given VMPL to modify permissions for a less privileged VMPL
        - restricted so one level cannot grant more permissions than it currently has
    - AMD-V 2-level paging1
      - translate a Guest Virtual Address (GVA) to a Guest Physical Address (GPA)
        - to a System Physical Address (SPA).
      - every SPA can only be mapped to a single GPA at one time.
      - each RMP entry is a Validated bit
      - 
  - Virtual Machine Privilege Levels
    - provide hardware isolated abstraction layers within a VM for additional security controls
    - assistance with managing communication with the hypervisor
    - RMPADJUST
    - allows a guest VM to divide its address space into four levels
    - primarily used to set additional page permission checks and are otherwise orthogonal to other x86 security features
- memory integrity protection
  - data replay
  - corruption
  - memory re-mapping
  - aliasing based attacks
  - if a VM is able to read a private (encrypted) page of memory, it must always read the value it last wrote
- stronger protection around interrupt behavior
- increased protection against recently disclosed side channel attacks
- requires a combination of new CPU hardware and firmware
- shared (unencrypted) memory

- SEV-SNP-strengthening-vm-isolation-with-integrity-protection-and-more-1e2d0f1f63c9168a575bd487c8c8698b
  - Secure Encrypted Virtualization (SEV) 3
    - minimize their exposure to bugs in the cloud provider's infrastructure 3
    - keep their data confidential 3
    - isolate VMs at a hardware level 3
      - individual VMs could be assigned a unique AES encryption key 3
  - SEV-ES (Encrypted State) 3
    - protection for CPU register state 3
  - SEV-SNP (Secure Nested Paging) 3
    - memory integrity protection 3
      - data replay 3
      - corruption 6
      - memory re-mapping 3
      - aliasing based attacks 6
      - if a VM is able to read a private (encrypted) page of memory, it must always read the value it last wrote 4
    - stronger protection around interrupt behavior 3
    - increased protection against recently disclosed side channel attacks 3
    - requires a combination of new CPU hardware and firmware 4
    - Threat Model 5

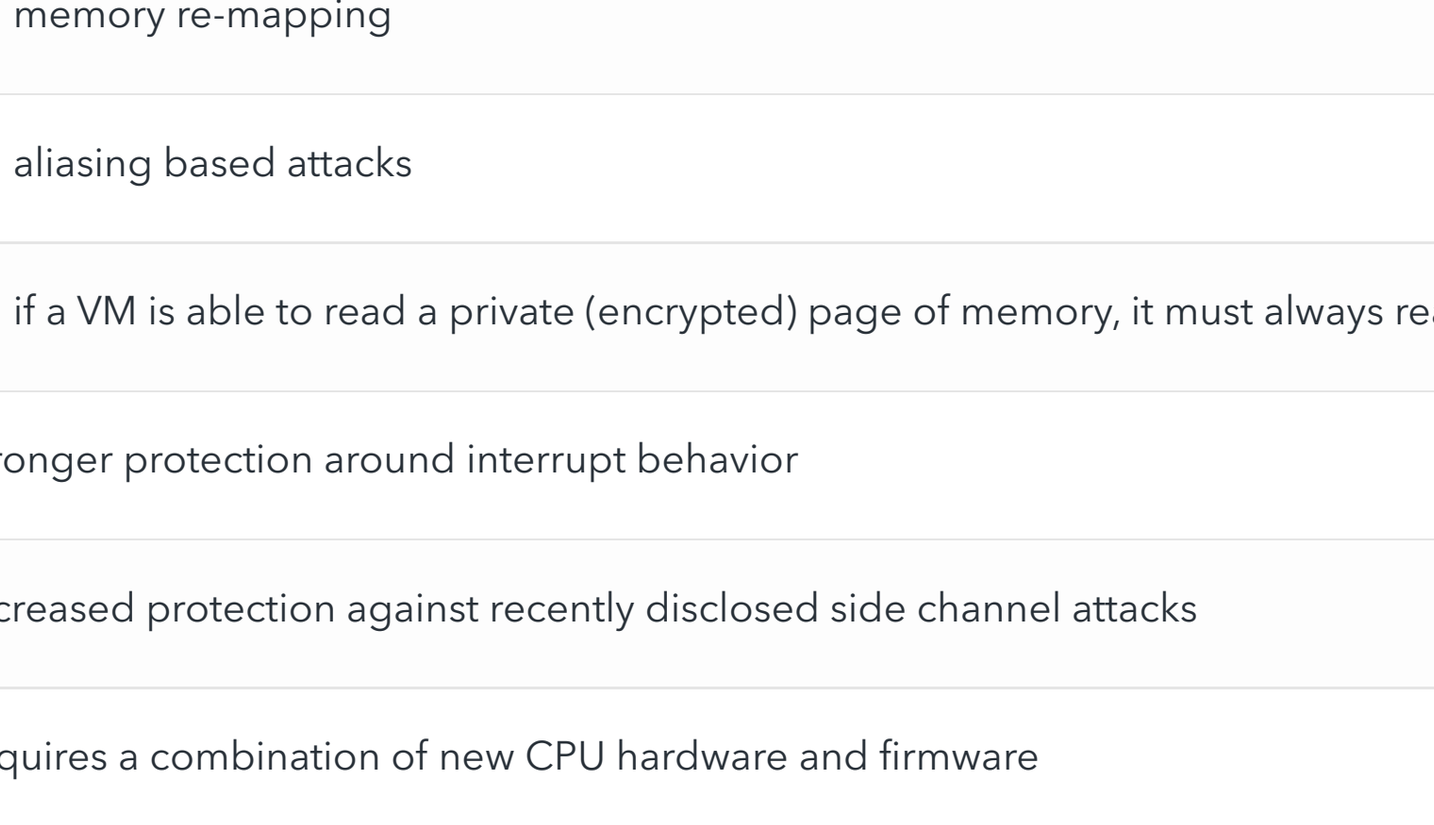


FIGURE 2: SEV-SNP THREAT MODEL

5

- fully trusted — 5
  - AMD Secure Processor (AMD-SP) — 5
  - VM — 5
- fully untrusted — 5
  - all other CPU software components — 5
    - BIOS on the host system — 5
    - hypervisor — 5
    - device drivers — 5
    - other VMs — 5
  - PCI devices — 5
- SEV technologies only protect data in-use — 5
- ✓ = Mitigated ★ = Optionally Mitigated ⊘ = Not Mitigated — 7

Potential Threats	*SEV*	*SEV-ES*	*SEV-SNP*
Confidentiality			
VM Memory *Example attack: Hypervisor reads private VM memory*	✓	✓	✓
VM Register State *Example attack: Read VM register state after VMEXIT*	⊘	✓	✓
DMA Protection *Example attack: Device attempts to read VM memory*	✓	✓	✓
Integrity			
Replay Protection *Example attack: Replace VM memory with an old copy*	⊘	⊘	✓
Data Corruption *Example attack: Replace VM memory with junk data*	⊘	⊘	✓
Memory Aliasing *Example attack: Map two guest pages to same DRAM page*	⊘	⊘	✓
Memory Re-Mapping *Example attack: Switch DRAM page mapped to a guest page*	⊘	⊘	✓
Availability			
Denial of Service on Hypervisor *Example attack: Malicious guest refuses to yield/exit*	✓	✓	✓
Denial of Service on Guest *Example attack: Malicious hypervisor refuses to run guest*	⊘	⊘	⊘
Physical Access Attacks			
Offline DRAM analysis *Example attack: Cold boot*	✓	✓	✓
Active DRAM corruption *Example attack: Manipulate DDR bus while VM is running*	⊘	⊘	⊘
Misc.			
TCB Rollback *Example attack: Revert AMD-SP firmware to old version*	⊘	⊘	✓
Malicious Interrupt/Exception Injection *Example attack: Inject interrupt while RFLAGS.IF=0*	⊘	⊘	★
Indirect Branch Predictor Poisoning *Example attack: Poison BTB from hypervisor*	⊘	⊘	★
Secure Hardware Debug Registers *Example attack: Change breakpoints during debug*	⊘	⊘	★
Trusted CPUID Information *Example attack: Hypervisors lies about platform capabilities*	⊘	⊘	★
Architectural Side Channels *Example attack: PRIME+PROBE to track VM accesses*	⊘	⊘	⊘
Page-level Side Channels *Example attack: Track VM access patterns through page tables*	⊘	⊘	⊘
Performance Counter Tracking *Example attack: Fingerprint VM apps by performance data*	⊘	⊘	⊘

TABLE 1: THREAT MODEL

- Reverse Map Table (8)
  - single data structure shared across the system (8)
  - one entry for every 4k page of DRAM that may be used by VMs (8)
  - tracks the owner for each page of memory (8)
  - in conjunction with standard x86 page tables (8)
  - New CPU instructions exist to enable manipulation of RMP entries (9)
- PVALIDATE (10)
  - sets the Validated bit (10)
- RMPUPDATE (10)
  - assigns the page to the guest (10)
- (10)

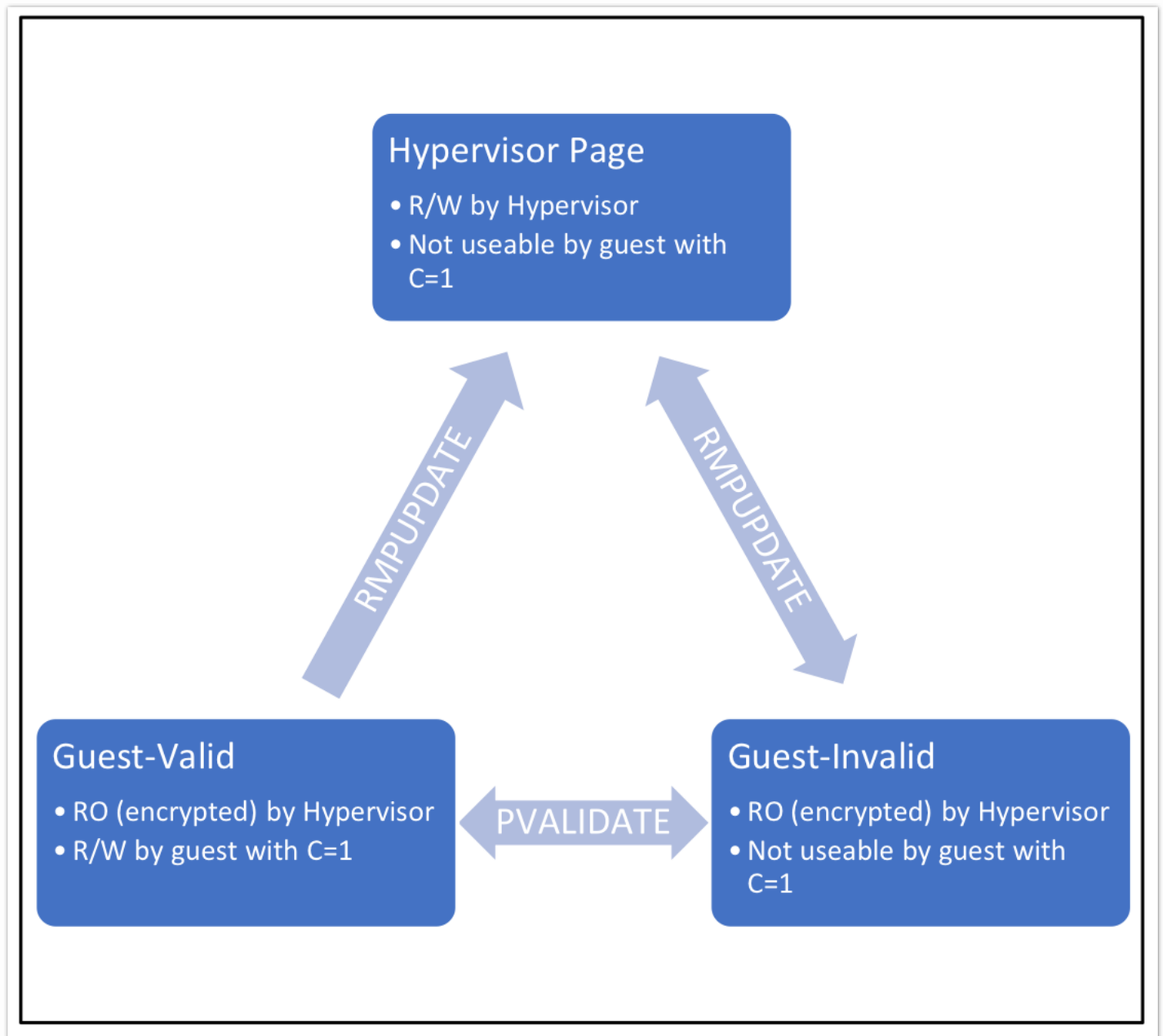


FIGURE 4: BASIC PAGE STATES

- RMPADJUST (13)
  - allows a given VMPL to modify permissions for a less privileged VMPL (13)
  - restricted so one level cannot grant more permissions than it currently has (13)
- AMD-V 2-level paging1 (9)

- translate a Guest Virtual Address (GVA) to a Guest Physical Address (GPA) — 9
  - to a System Physical Address (SPA). — 9
- every SPA can only be mapped to a single GPA at one time. — 10
- each RMP entry is a Validated bit — 10
- 11

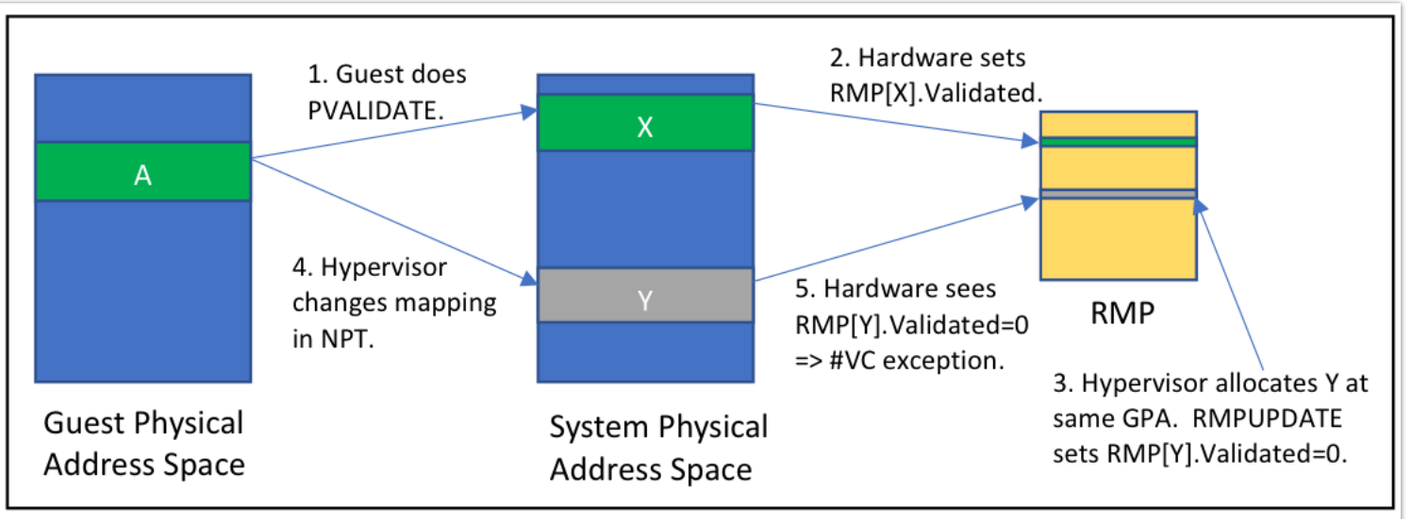


FIGURE 5: PAGE RE-MAPPING ATTACK

- Virtual Machine Privilege Levels — 13
  - provide hardware isolated abstraction layers within a VM for additional security controls — 13
  - assistance with managing communication with the hypervisor — 13
  - RMPADJUST — 13
  - allows a guest VM to divide its address space into four levels — 13
  - primarily used to set additional page permission checks and are otherwise orthogonal to other x86 security features — 13
- shared (unencrypted) memory — 4