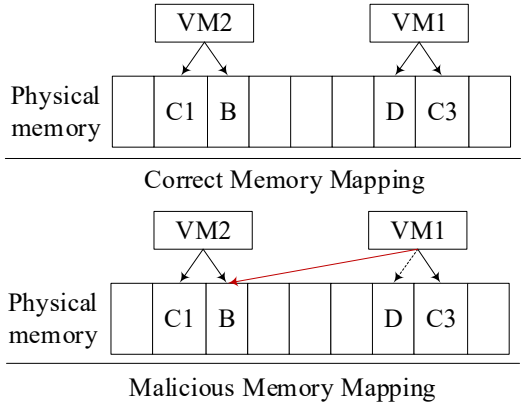
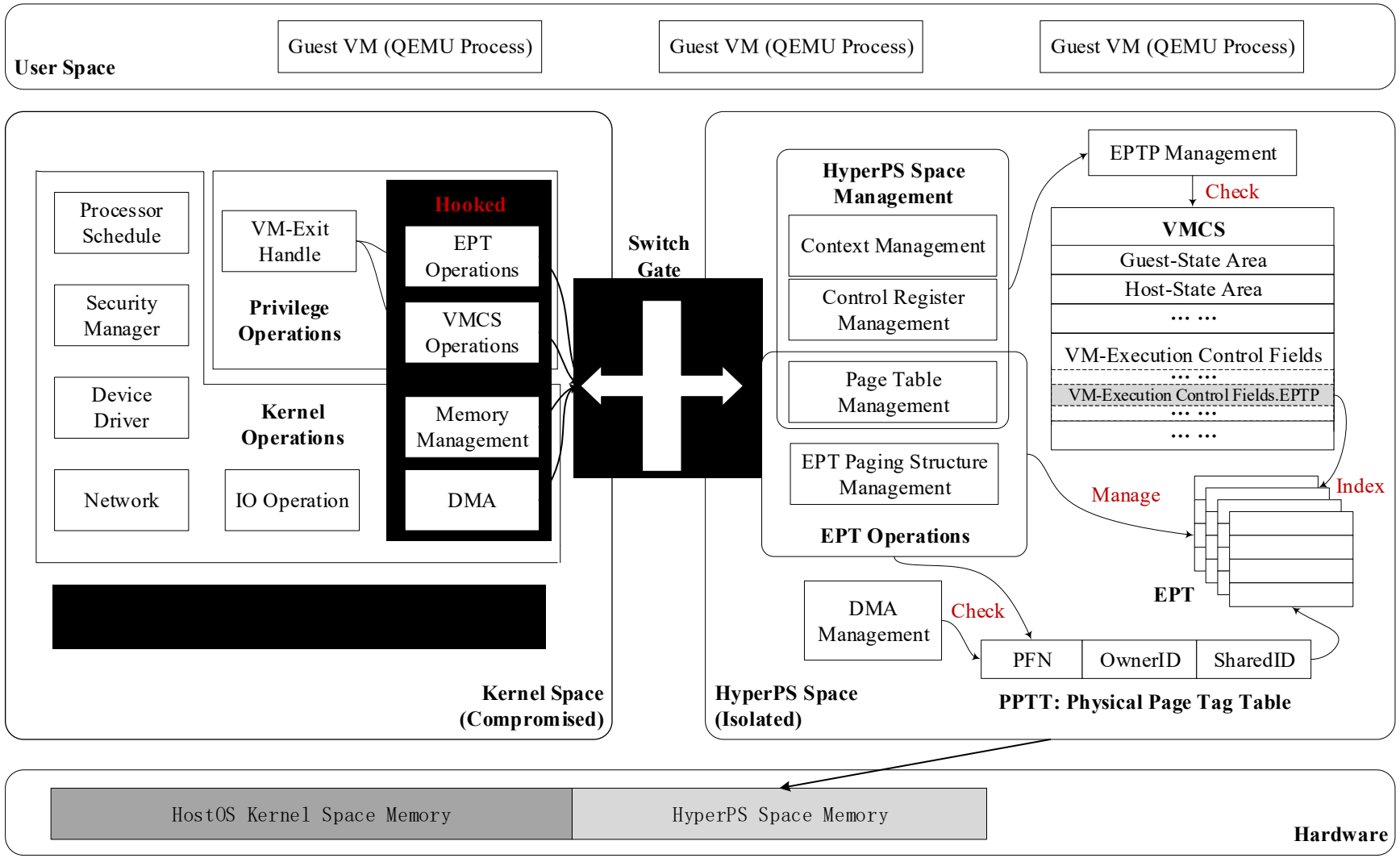
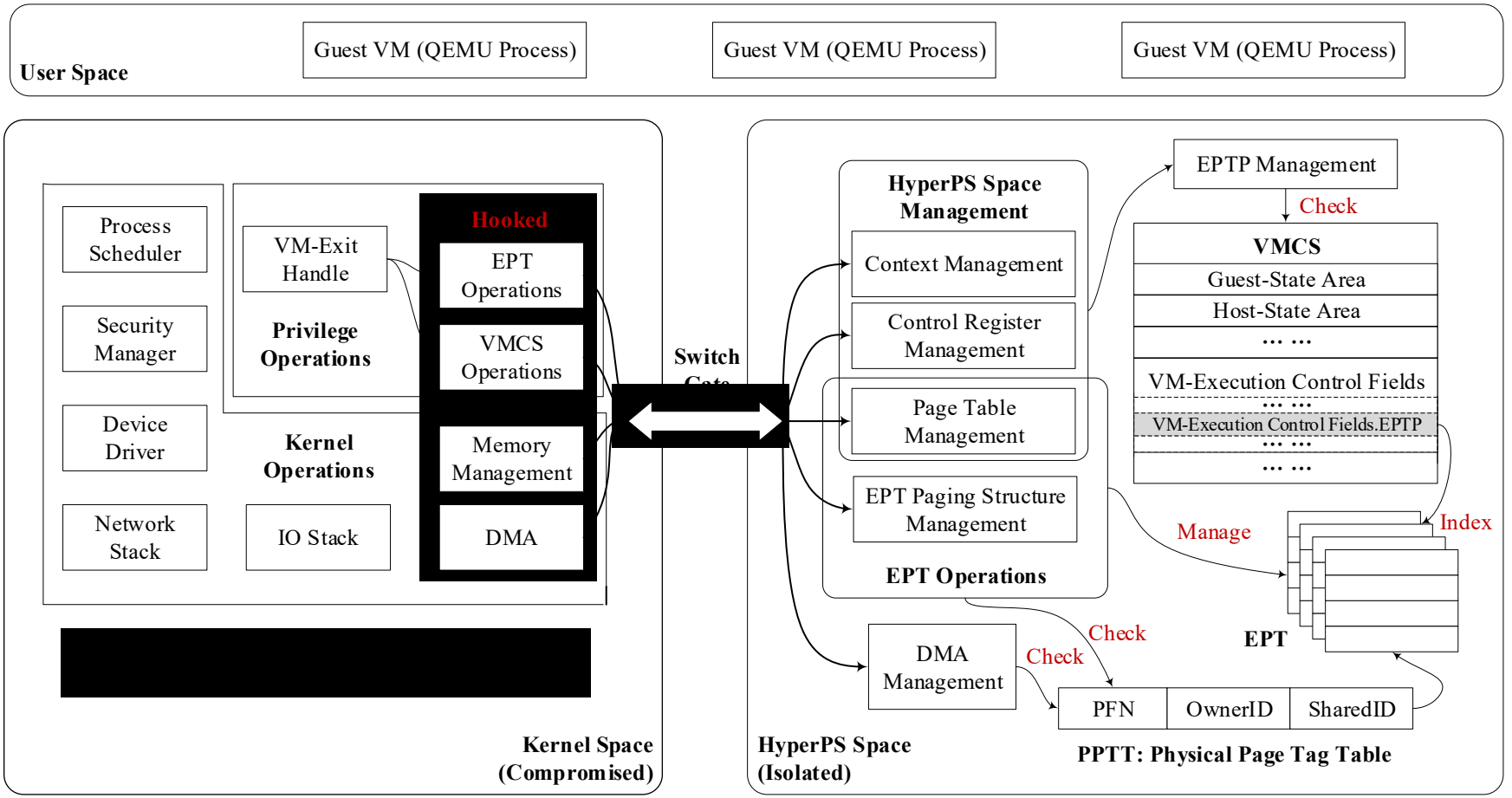
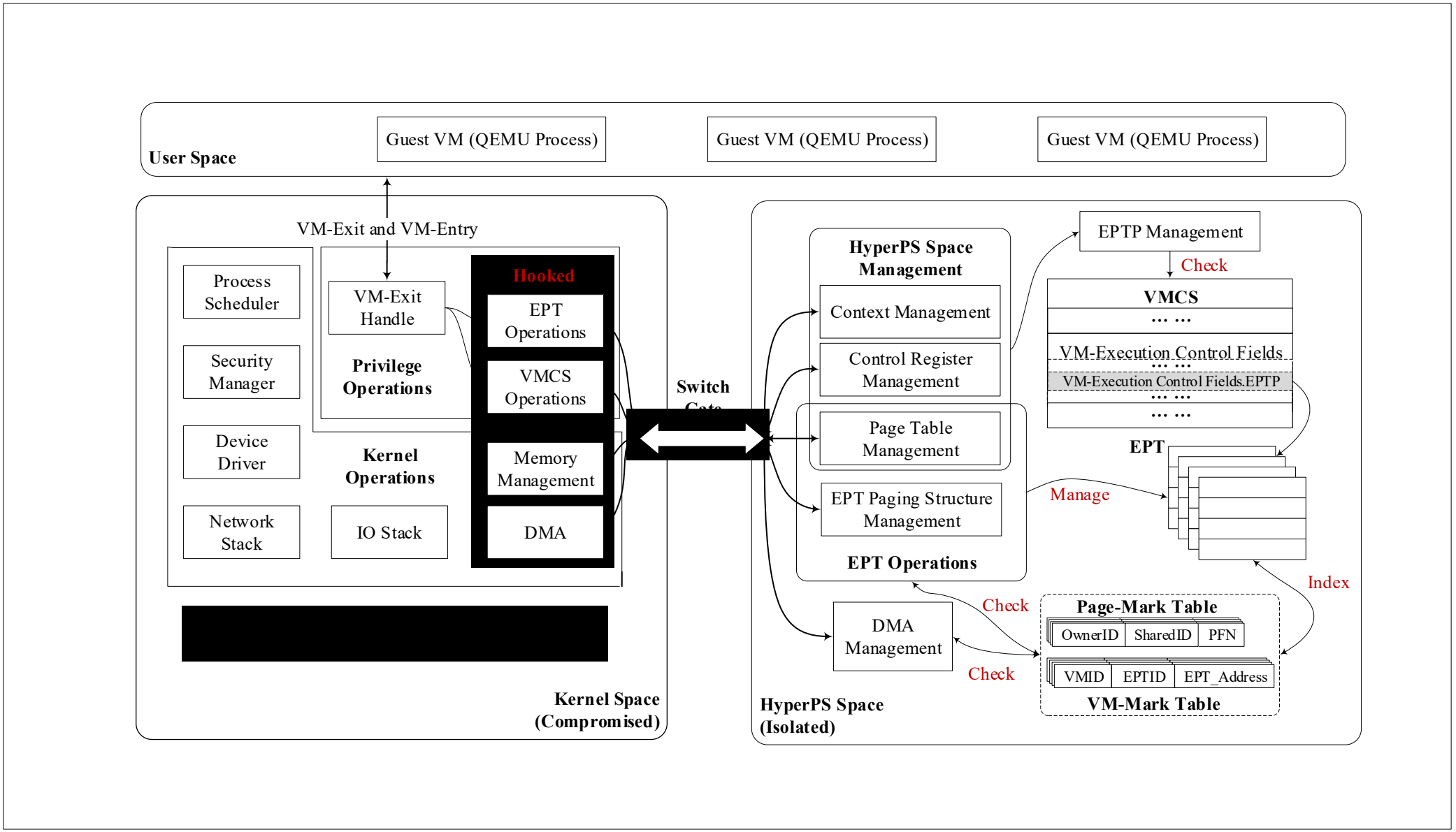
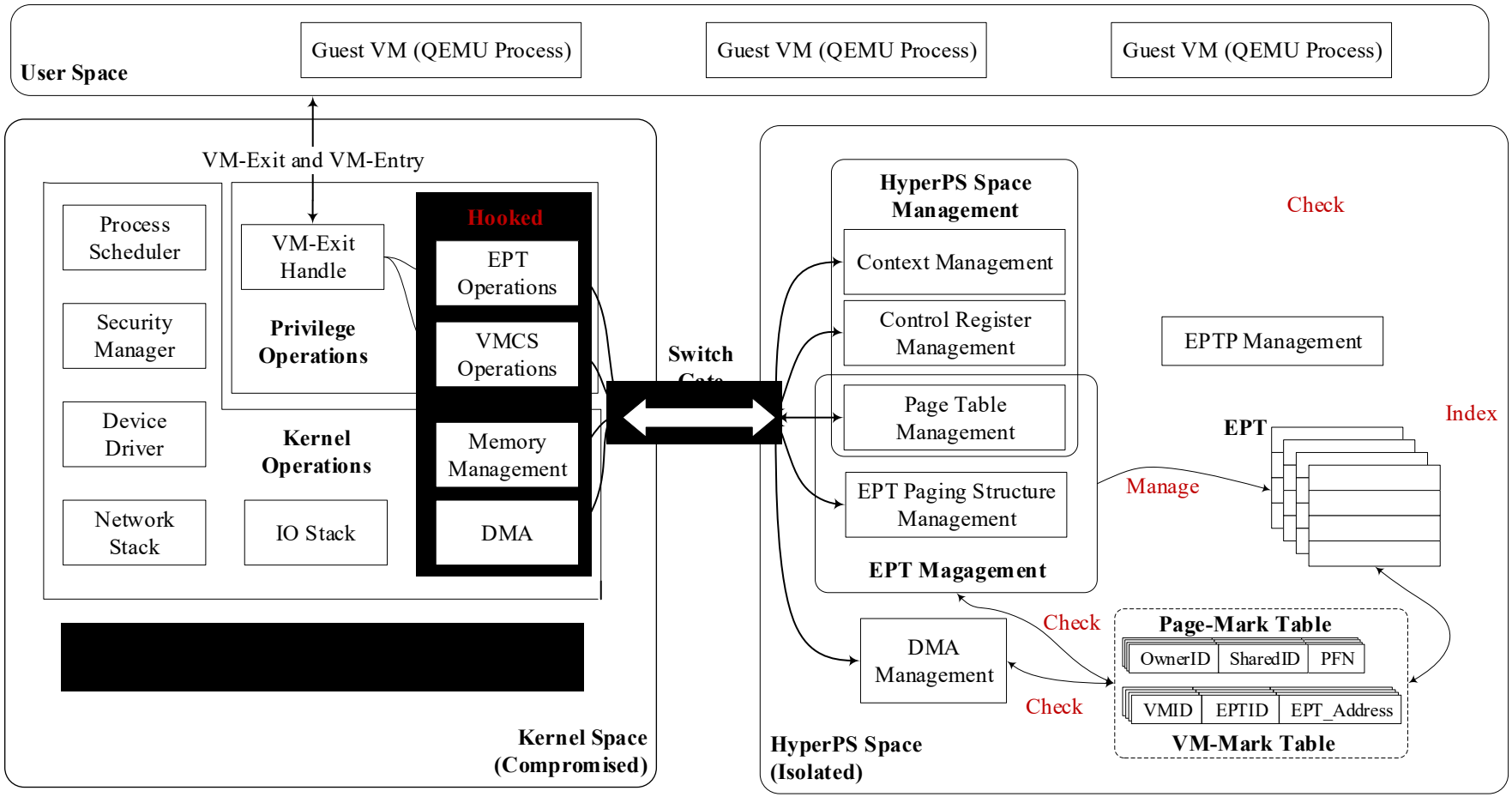

VM2
VM1
Physical memory
C1
B
D
C3
Correct Memory Mapping
VM2
VM1
Physical memory
C1
B
D
C3
Malicious Memory Mapping

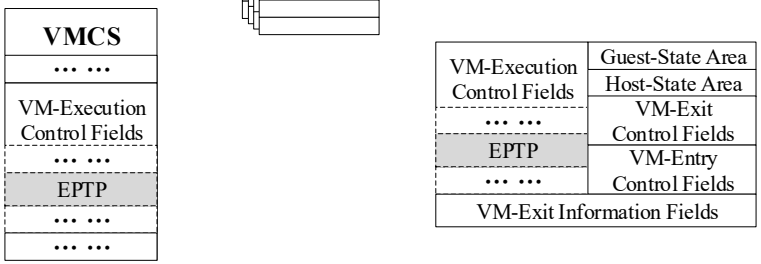

User Space
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Processor Schedule
Security Manager
Device Driver
Network
VM-Exit Handle
Privilege Operations
Kernel Operations
IO Operation
Hooked
EPT Operations
VMCS Operations
Memory Management
DMA
Switch Gate
Kernel Space (Compromised)
HyperPS Space Management
Context Management
Control Register Management
Page Table Management
EPT Paging Structure Management
EPT Operations
EPTP Management
Check
VMCS
Guest-State Area
Host-State Area
VM-Execution Control Fields
VM-Execution Control Fields.EPTP
Manage
Index
EPT
DMA Management
Check
PFN
OwnerID
SharedID
PPTT: Physical Page Tag Table
HyperPS Space (Isolated)
HostOS Kernel Space Memory
HyperPS Space Memory
Hardware


User Space
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Process Scheduler
Security Manager
Device Driver
Network Stack
VM-Exit Handle
Privilege Operations
Kernel Operations
IO Stack
Hooked
EPT Operations
VMCS Operations
Memory Management
DMA
Switch
Kernel Space (Compromised)
HyperPS Space Management
Context Management
Control Register Management
Page Table Management
EPT Paging Structure Management
EPT Operations
EPTP Management
Check
VMCS
Guest-State Area
Host-State Area
VM-Execution Control Fields
VM-Execution Control Fields.EPTP
Manage
Index
EPT
DMA Management
Check
Check
PFN
OwnerID
SharedID
PPTT: Physical Page Tag Table
HyperPS Space (Isolated)

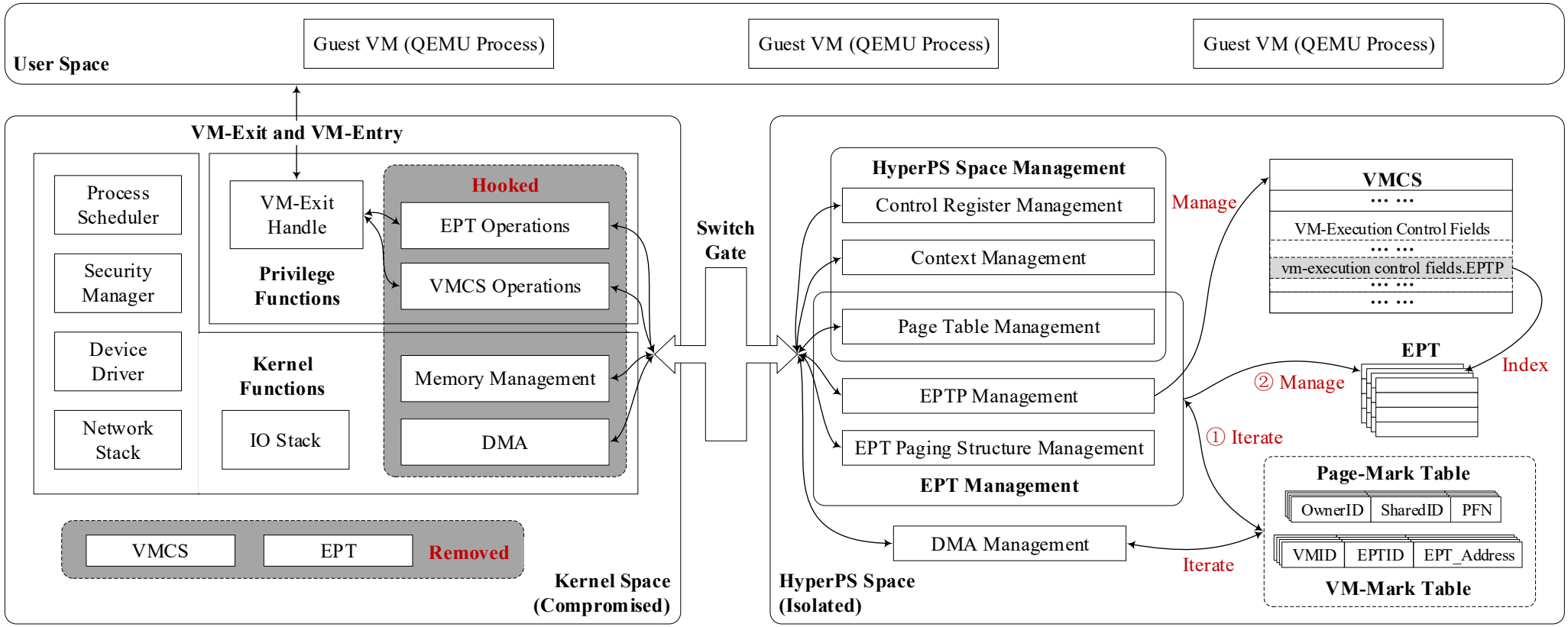

User Space
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Guest VM (QEMU Process)
VM-Exit and VM-Entry
Process Scheduler
Security Manager
Device Driver
Network Stack
VM-Exit Handle
Privilege Operations
Hooked
EPT Operations
VMCS Operations
Memory Management
DMA
Kernel Operations
IO Stack
Switch
Kernel Space (Compromised)
HyperPS Space Management
Context Management
Control Register Management
Page Table Management
EPT Paging Structure Management
EPT Operations
EPTP Management
Check
VMCS
VM-Execution Control Fields
VM-Execution Control Fields.EPTP
EPT
Manage
Index
DMA Management
Check
Check
Page-Mark Table
OwnerID
SharedID
PFN
VMID
EPTID
EPT_Address
VM-Mark Table
HyperPS Space (Isolated)

V3

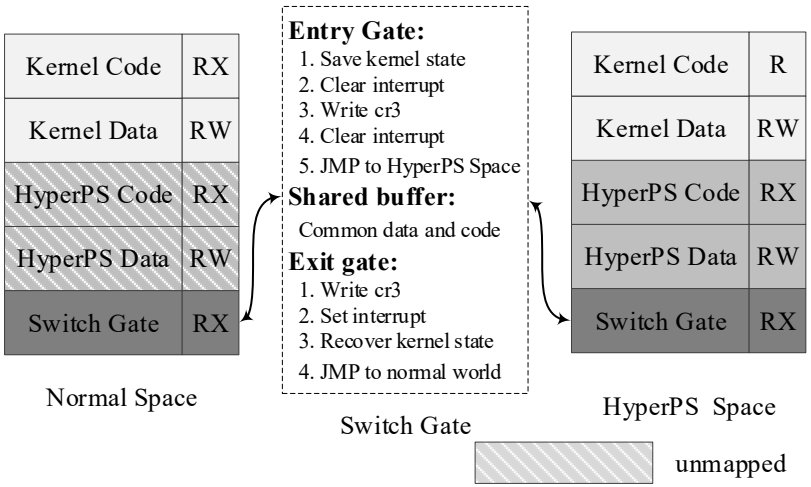

User Space
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Guest VM (QEMU Process)
VM-Exit and VM-Entry
Process Scheduler
Security Manager
Device Driver
Network Stack
VM-Exit Handle
Privilege Operations
Hooked
EPT Operations
VMCS Operations
Memory Management
DMA
Kernel Operations
IO Stack
Switch
Kernel Space (Compromised)
HyperPS Space Management
Context Management
Control Register Management
Page Table Management
EPT Paging Structure Management
EPT Magagement
Check
EPTP Management
Index
EPT
Manage
DMA Management
Check
Check
Page-Mark Table
OwnerID
SharedID
PFN
VMID
EPTID
EPT_Address
VM-Mark Table
HyperPS Space (Isolated)

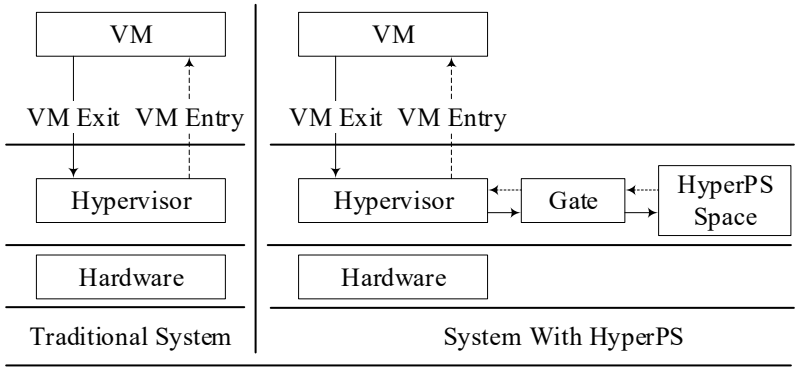

VMCS
VM-Execution Control Fields
EPTP
Guest-State Area
Host-State Area
VM-Exit Control Fields
VM-Entry Control Fields
VM-Exit Information Fields

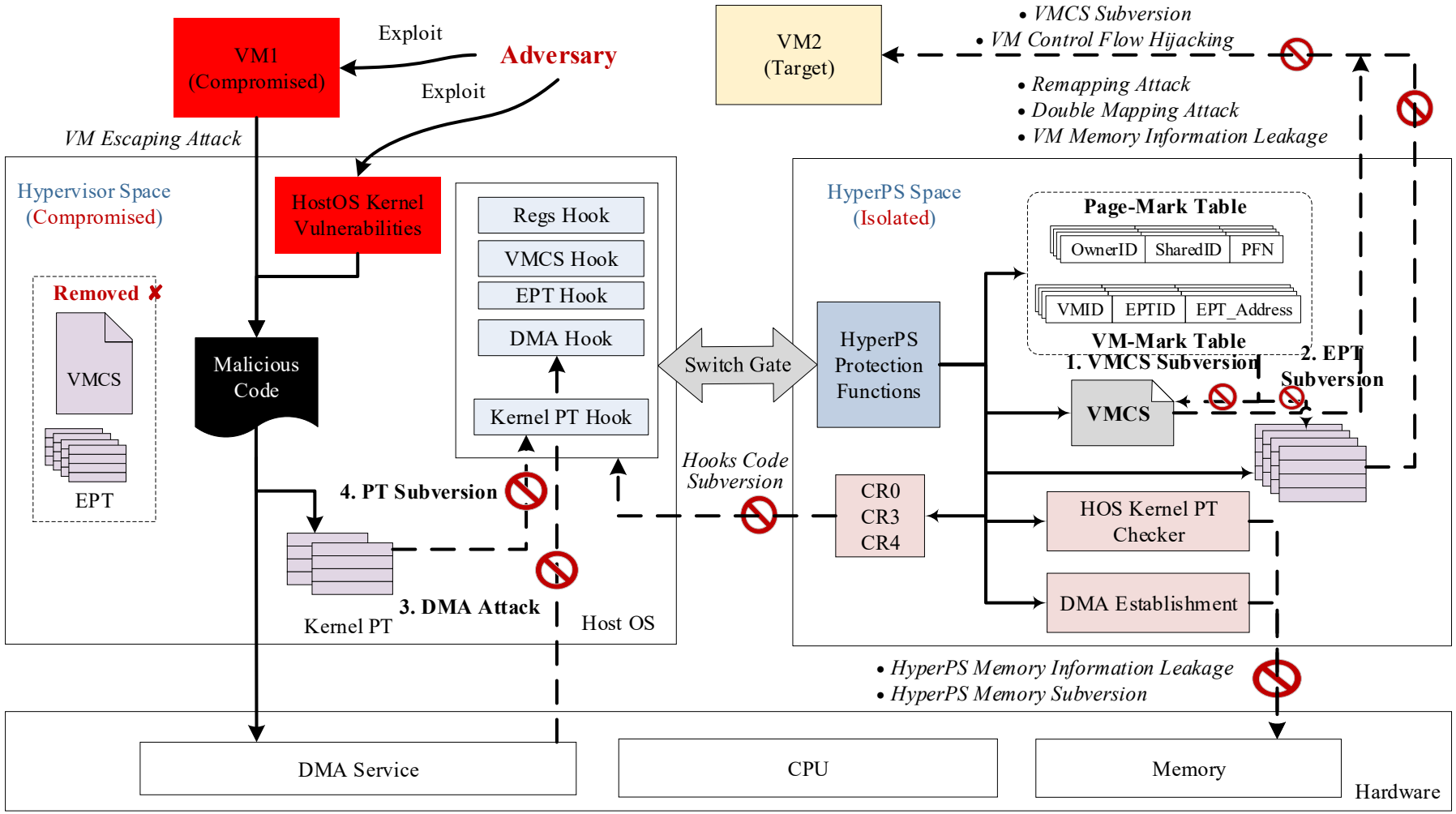
V4

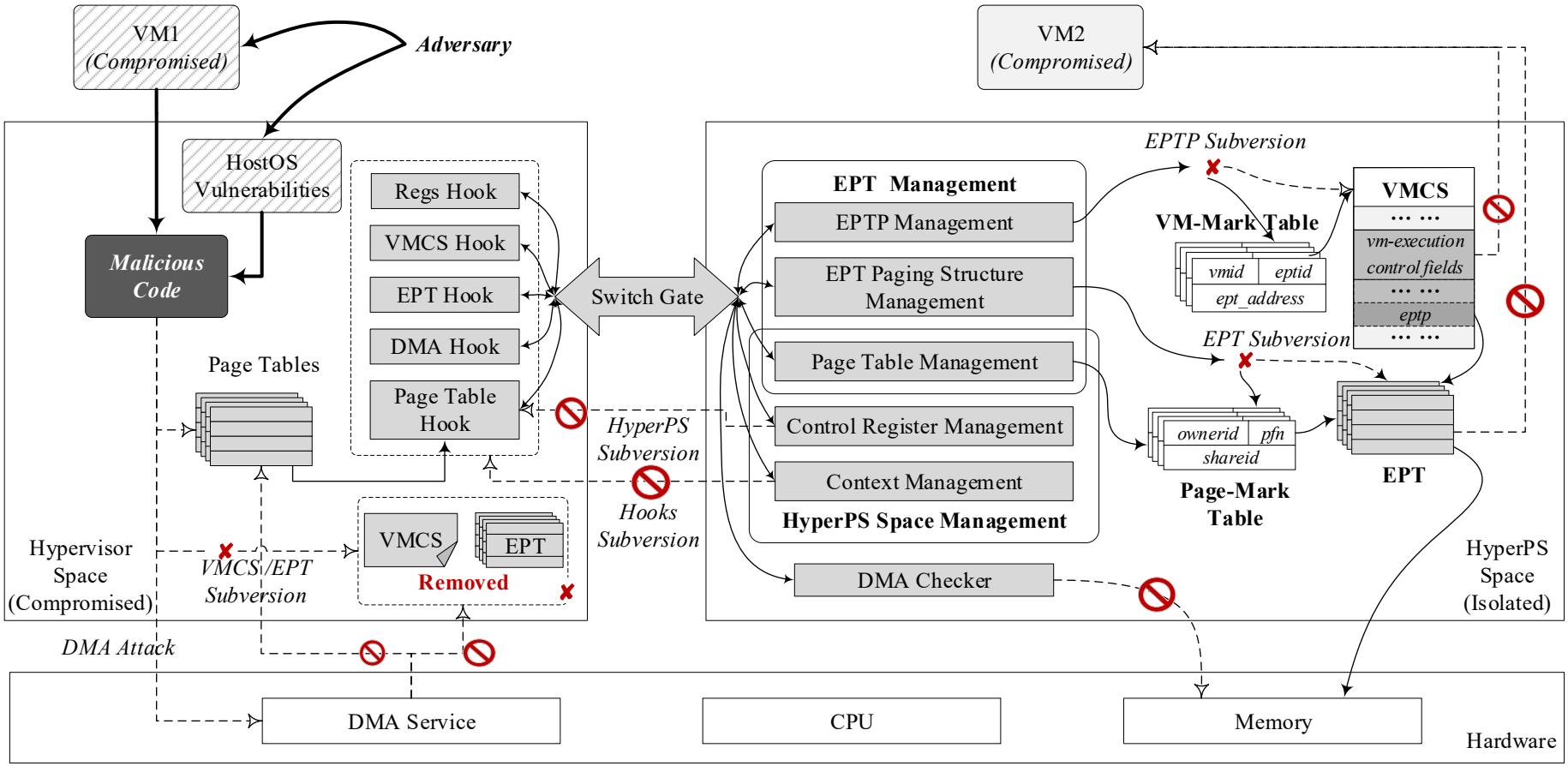

User Space
Guest VM (QEMU Process)
Guest VM (QEMU Process)
Guest VM (QEMU Process)
VM-Exit and VM-Entry
Process Scheduler
Security Manager
Device Driver
Network Stack
VM-Exit Handle
Privilege Functions
Hooked
EPT Operations
VMCS Operations
Kernel Functions
Memory Management
IO Stack
DMA
VMCS
EPT
Removed
Kernel Space (Compromised)
Switch Gate
HyperPS Space Management
Control Register Management
Context Management
Page Table Management
EPTP Management
EPT Paging Structure Management
EPT Management
DMA Management
HyperPS Space (Isolated)
Manage
VMCS
VM-Execution Control Fields
vm-execution control fields.EPTP
EPT
Index
② Manage
① Iterate
Page-Mark Table
OwnerID
SharedID
PFN
VMID
EPTID
EPT_Address
VM-Mark Table
Iterate

V5


Kernel Code
RX
Kernel Data
RW
HyperPS Code
RX
HyperPS Data
RW
Switch Gate
RX
Normal Space
Entry Gate:
1. Save kernel state
2. Clear interrupt
3. Write cr3
4. Clear interrupt
5. JMP to HyperPS Space
Shared buffer:
Common data and code
Exit gate:
1. Write cr3
2. Set interrupt
3. Recover kernel state
4. JMP to normal world
Switch Gate
Kernel Code
R
Kernel Data
RW
HyperPS Code
RX
HyperPS Data
RW
Switch Gate
RX
HyperPS Space
unmapped


VM
VM Exit
VM Entry
Hypervisor
Hardware
Traditional System
VM
VM Exit
VM Entry
Hypervisor
Gate
HyperPS Space
Hardware
System With HyperPS

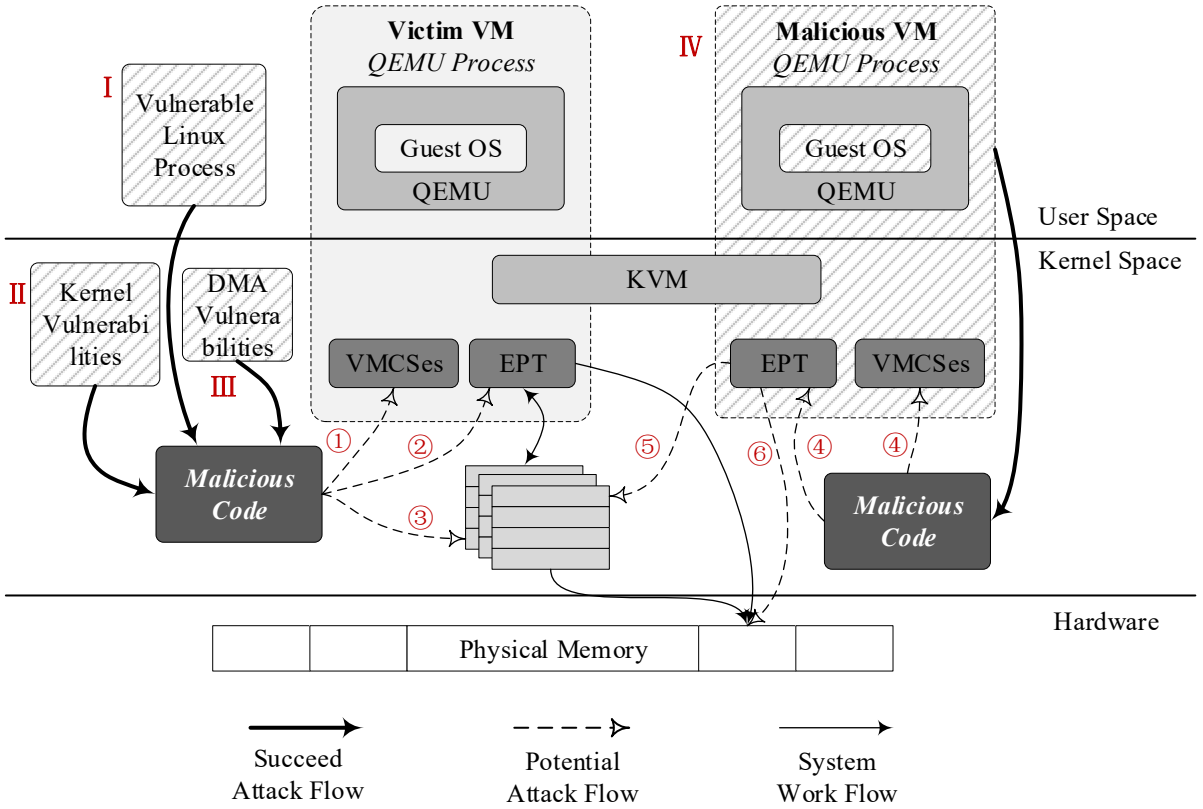

VM1
(Compromised)
Exploit
Adversary
Exploit
VM2
(Target)
• VMCS Subversion
• VM Control Flow Hijacking
• Remapping Attack
• Double Mapping Attack
• VM Memory Information Leakage
VM Escaping Attack
Hypervisor Space
(Compromised)
HostOS Kernel
Vulnerabilities
Regs Hook
VMCS Hook
EPT Hook
DMA Hook
Kernel PT Hook
HyperPS Space
(Isolated)
Page-Mark Table
OwnerID
SharedID
PFN
VMID
EPTID
EPT_Address
VM-Mark Table
Removed ✘
VMCS
EPT
Malicious
Code
Switch Gate
HyperPS
Protection
Functions
1. VMCS Subversion
2. EPT
Subversion
VMCS
Hooks Code
Subversion
4. PT Subversion
CR0
CR3
CR4
HOS Kernel PT
Checker
DMA Establishment
Kernel PT
3. DMA Attack
Host OS
• HyperPS Memory Information Leakage
• HyperPS Memory Subversion
DMA Service
CPU
Memory
Hardware

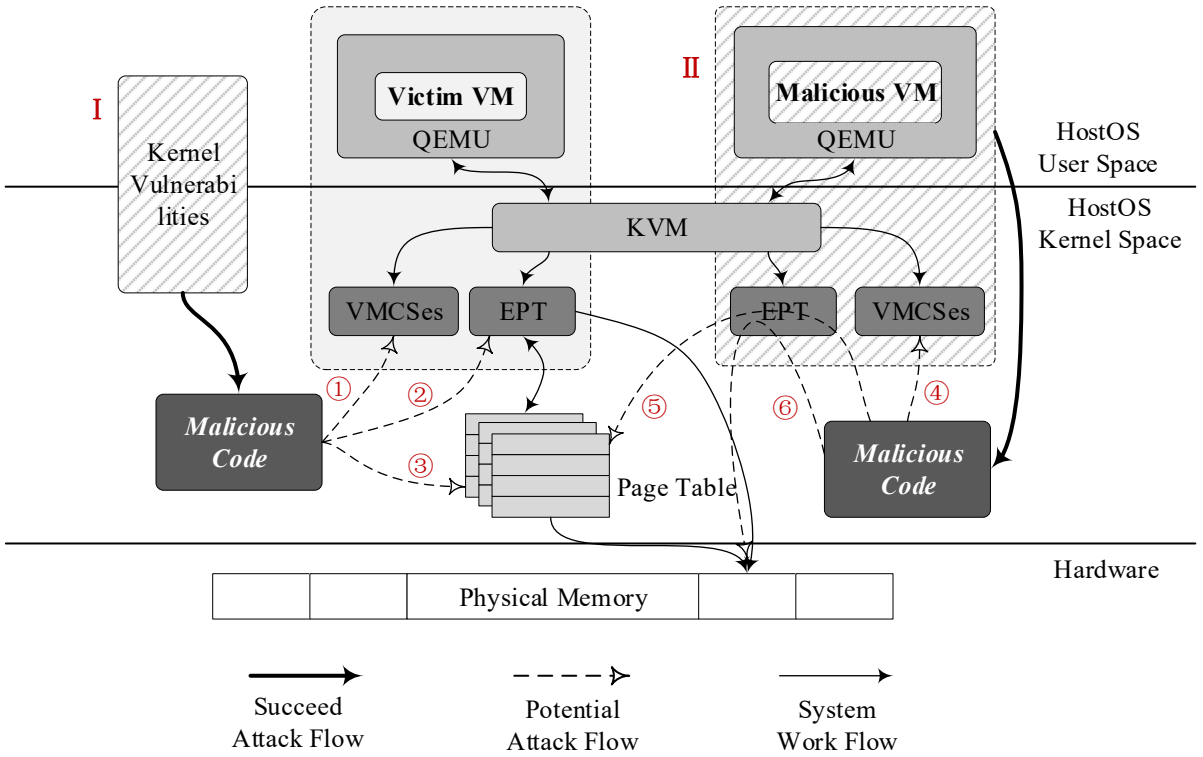


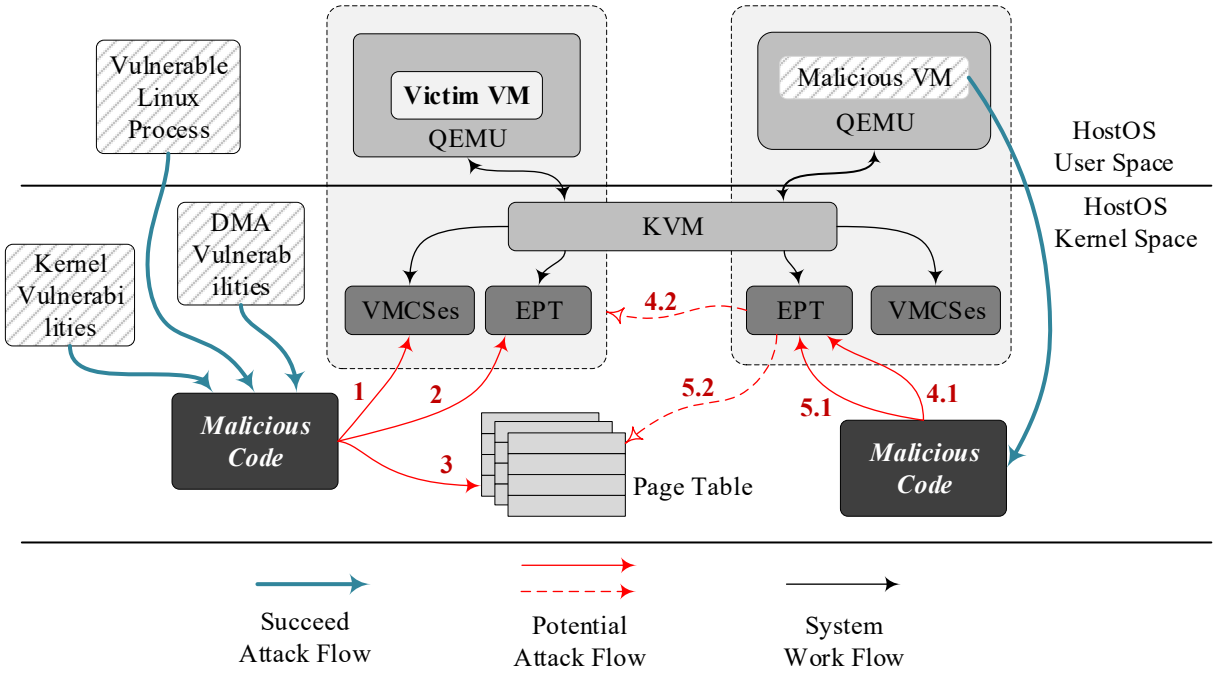
VMCS	
guest-state	vm-execution control
host-state	
vm-entry control	eptp
vm-exit control	vm-exit information

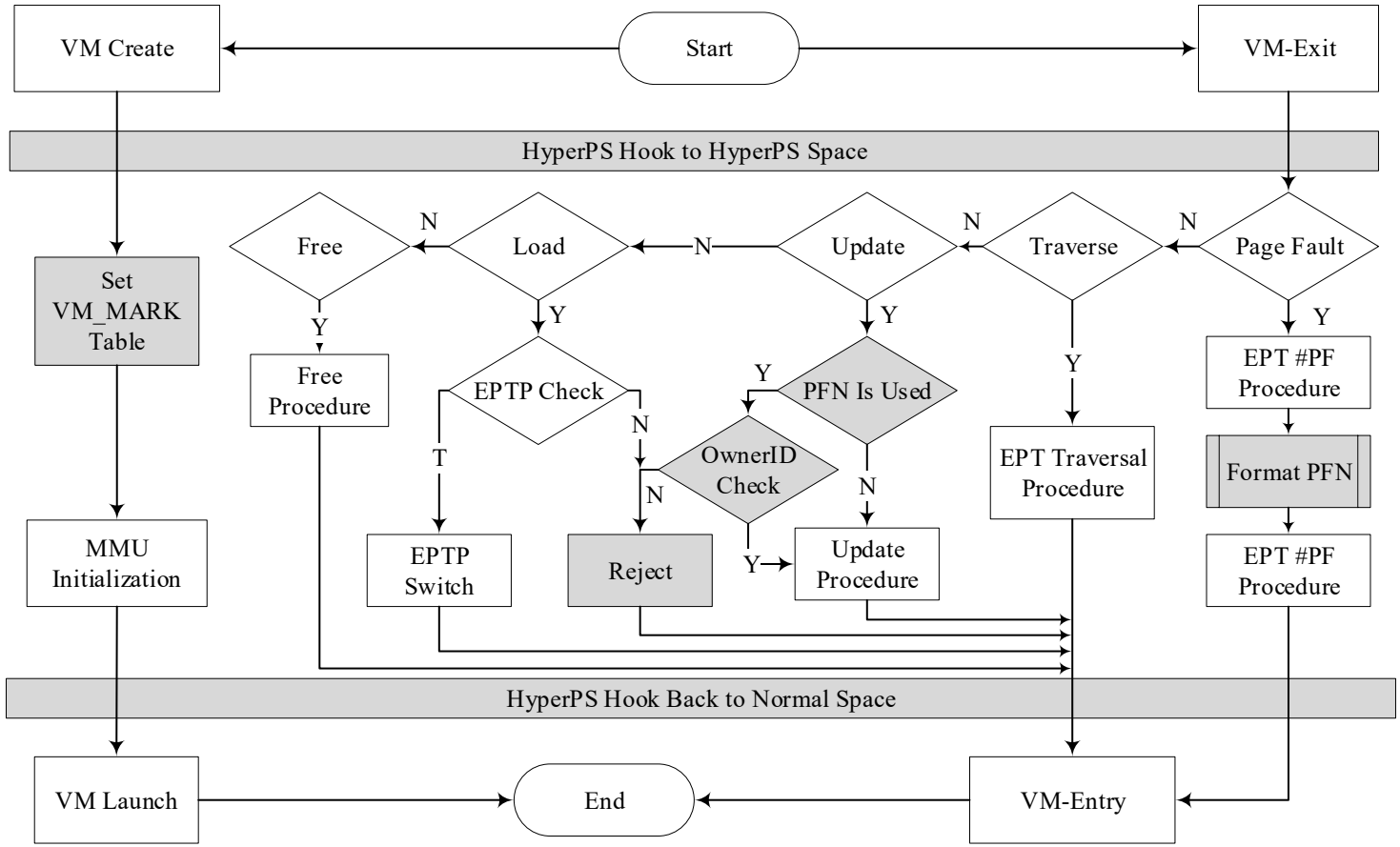
VMCS	
… …	vm-execution control fields
… …	… …
… …	eptp
… …	… …

DMA Management


Victim VM
QEMU Process
Guest OS
QEMU
Malicious VM
QEMU Process
Guest OS
QEMU
I
Vulnerable Linux Process
II
Kernel Vulnerabilities
DMA Vulnerabilities
III
IV
KVM
VMCSes
EPT
EPT
VMCSes
User Space
Kernel Space
Malicious Code
Malicious Code
①
②
③
④
④
⑤
⑥
Hardware
Physical Memory
Succeed Attack Flow
Potential Attack Flow
System Work Flow


I
Kernel
Vulnerabi
lities
Victim VM
QEMU
II
Malicious VM
QEMU
HostOS
User Space
HostOS
Kernel Space
KVM
VMCSes
EPT
EPT
VMCSes
①
②
③
④
⑤
⑥
Malicious
Code
Page Table
Malicious
Code
Hardware
Physical Memory
Succeed
Attack Flow
Potential
Attack Flow
System
Work Flow


Vulnerable Linux Process
Victim VM
QEMU
Malicious VM
QEMU
HostOS User Space
HostOS Kernel Space
KVM
DMA Vulnerab ilities
Kernel Vulnerabi lities
VMCSes
EPT
EPT
VMCSes
4.2
1
2
3
5.2
5.1
4.1
Malicious Code
Page Table
Malicious Code
Succeed Attack Flow
Potential Attack Flow
System Work Flow


VM Create
Start
VM-Exit
HyperPS Hook to HyperPS Space
Free
Load
Update
Traverse
Page Fault
N
Y
Set VM_MARK Table
Free Procedure
EPTP Check
T
PFN Is Used
OwnerID Check
Update Procedure
EPT Traversal Procedure
EPT #PF Procedure
Format PFN
MMU Initialization
EPTP Switch
Reject
HyperPS Hook Back to Normal Space
VM Launch
End
VM-Entry