University of BRISTOL

Systems Security

COMSM1500

bristol.ac.uk

University of BRISTOL

Isolation

bristol.ac.uk

Plan

- Need for isolation
- Virtual Machines
- Containers
- Unikernels
- SGX Enclaves

Threats

- Vulnerable software
  - Contains design/implementation vulnerabilities that can be exploited
  - e.g. buffer overflow can lead to the execution of arbitrary code

Threats

- Vulnerable software
  - Contains design/implementation vulnerabilities that can be exploited
  - e.g. buffer overflow can lead to the execution of arbitrary code
- Malware
  - Software designed to act maliciously
  - e.g. fake “security issue” website that download software to fix it
    - My mom love those
  - Genuine software tool chain can be compromised
  - e.g. we mentioned a few lectures ago in China compromised SDK

Objective

- Those program execute locally with user privilege
- It is likely to happen at some point
  - e.g. remote code execution vulnerability discovered in flash every other month
- Can we reduce damage a piece of software can do?

Objective

- Those program execute locally with user privilege
- It is likely to happen at some point
  - e.g. remote code execution vulnerability discovered in flash every other month
- Can we reduce damage a piece of software can do?

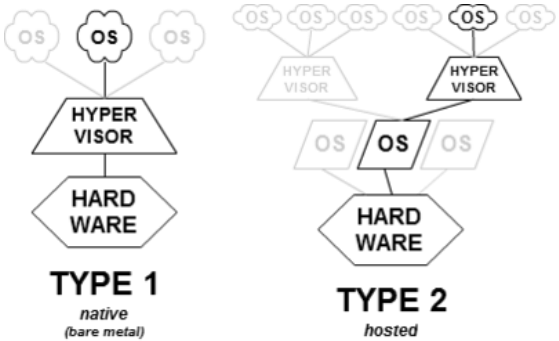
Homework/exam question:
Explain the motivation behind isolation/sandboxing

Isolation-based sandbox

- Run each application in an isolated sandbox environment
- Can only access resources within the sandbox

Isolation: system-level sandboxes

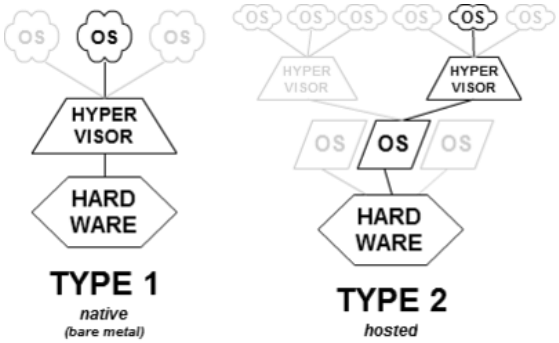
- System-level sandbox
  - Complete environment for an OS
- Virtualisation:
  - Hypervisor aka Virtual Machine Monitor
  - Multiplex hardware to run hardware-level virtual machines



Isolation: system-level sandboxes

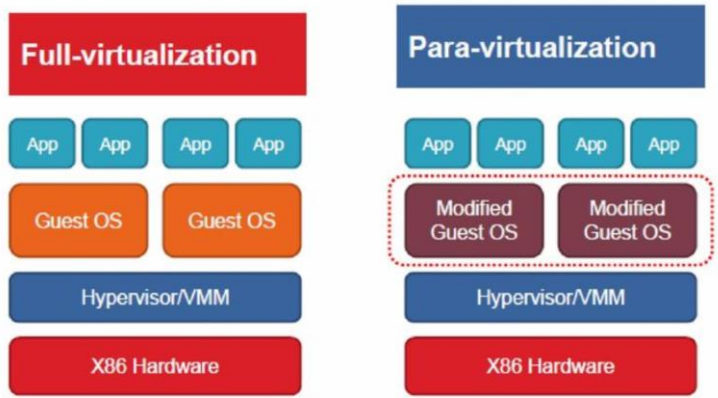
Homework/exam question:
Explain virtualization

- System-level sandbox
  - Complete environment for an OS
- Virtualisation:
  - Hypervisor aka Virtual Machine Monitor
  - Multiplex hardware to run hardware-level virtual machines



Isolation: system-level sandboxes

- Hardware virtualization
  - The guest does not need to know it is virtualized
  - e.g. VMWare, VirtualBox (some paravirtualization)
- Paravirtualization
  - The guest know it is virtualized
  - Can use specific API to increase performance
  - e.g. Xen



Isolation: system-level sandboxes

- Hardware virtualization
  - The guest does not need to know it is virtualized
  - e.g. VMWare, VirtualBox (some paravirtualization)
- Paravirtualization
  - The guest know it is virtualized
  - Can use specific API to increase performance
  - e.g. Xen
- Check the Xen paper on github (must read paper in CS)

University of BRISTOL

How does it help with security?

bristol.ac.uk

How does it help with security?

- Cloud computing context
- One application is compromised does not affect the other
  - Each OS has its own entire OS stack

How does it help with security?

Homework/exam question:
Discuss how virtualization may help with security

- Cloud computing context
- One application is compromised does not affect the other
  - Each OS has its own entire OS stack
- Or does it?

VMs share hardware

- VMs run on the same physical machine

VMs share hardware

- VMs run on the same physical machine
- Which means shared Cache and Instructions pipeline
  - … we saw in the second lecture that this is generally vulnerable
  - Can be exploited to extract information across VM
    - e.g. cryptographic keys

VMs share hardware

- VMs run on the same physical machine
- Which means shared Cache and Instructions pipeline
  - … we saw in the second lecture that this is generally vulnerable
  - Can be exploited to extract information across VM
    - e.g. cryptographic keys
- Which means performance measurement
  - allows to measure another VM activity
  - … we saw it can be used to infer (a lot of) information

VMs share hardware

- VMs run on the same physical machine
- Which means shared Cache and Instructions pipeline
  - … we saw in the second lecture that this is generally vulnerable
  - Can be exploited to extract information across VM
    - e.g. cryptographic keys
- Which means performance measurement
  - allows to measure another VM activity
  - … we saw it can be used to infer (a lot of) information
- Which means potential DoS
  - If the system is not carefully design I could starve other VMs

VMs share hardware

- Attack themselves may not be trivial
- … but, also need to ensure co-tenancy in cloud context
  - i.e. given a target attacker needs to run on the same physical machine

VMs share hardware

- Attack themselves may not be trivial
- … but, also need to ensure co-tenancy in cloud context
  - i.e. given a target attacker needs to run on the same physical machine
- Experiment done on Amazon EC2
- Can detect co-residency through network information

VMs share hardware

- Attack themselves may not be trivial
- … but, also need to ensure co-tenancy in cloud context
  - i.e. given a target attacker needs to run on the same physical machine
- Experiment done on Amazon EC2
- Can detect co-residency through network information
- Forcing co-residency
  - Brute force (8% success): i.e. launch instances until get a co-resident
  - Placement locality abuse (40% success)
    - VMs started around the same time are allocated the same hardware
    - Attacker can boot a lot of VM at the same time
    - Attacker can trigger victim VM to be allocated
      - e.g. triggering scaling, for webserver could be sending a lot of request to it

VMs share hardware

- Attack themselves may not be trivial
- … but, also need to ensure co-tenancy in cloud context
  - i.e. given a target attacker needs to run on the same physical machine
- Experiment done on Amazon EC2
- Can detect co-residency through network information
- Forcing co-residency
  - Brute force (8% success): i.e. launch instances until get a co-resident
  - Placement locality abuse (40% success)
    - VMs started around the same time are allocated the same hardware
    - Attacker can boot a lot of VM at the same time
    - Attacker can trigger victim VM to be allocated
      - e.g. triggering scaling, for webserver could be sending a lot of request to it
- Check on github “Hey, You, Get Off of My Cloud” paper

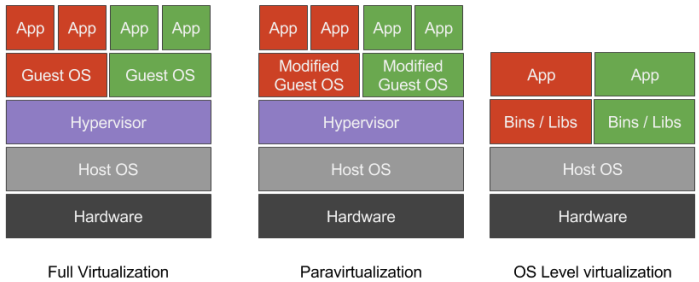
VMs share hardware

- Attack themselves may not be trivial
- … but, also need to ensure co-tenancy in cloud context
  - i.e. given a target attacker needs to run on the same physical machine
- Experiment done on Amazon EC2
- Can detect co-residency through network information
- Forcing co-residency
  - Brute force (8% success): i.e. launch instances until get a co-resident
  - Placement locality abuse (40% success)
    - VMs started around the same time are allocated the same hardware
    - Attacker can boot a lot of VM at the same time
    - Attacker can trigger victim VM to be allocated
      - e.g. triggering scaling, for webserver could be sending a lot of request to it
- Check on github “Hey, You, Get Off of My Cloud” paper

Homework/exam question: Discuss how you can steal information from a collocated virtual machine.

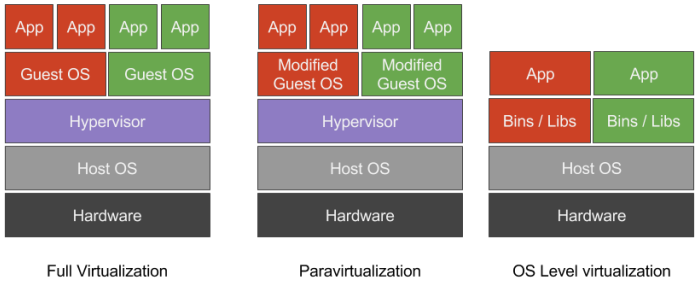
Container-based sandbox

- Share kernel but separate user space resources
- Performance wise more efficient



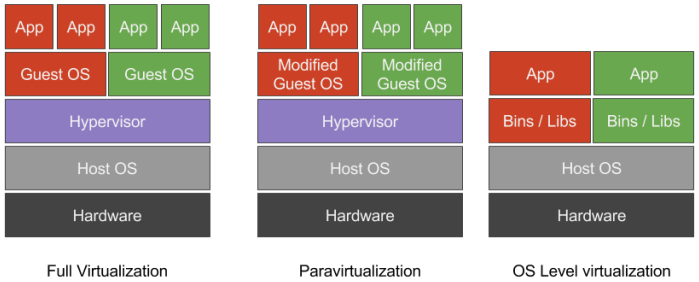
Container-based sandbox

- Share kernel but separate user space resources
- Performance wise more efficient
- … but concept of container do not really exist in the kernel
  - Use combination of features: namespaces, overlayfs etc…
  - Give the illusion of being “alone in the system”



Container-based sandbox

- Share kernel but separate user space resources
- Performance wise more efficient
- … but concept of container do not really exist in the kernel
  - Use combination of features: namespaces, overlayfs etc…
  - Give the illusion of being “alone in the system”



Homework/exam question: Discuss the kernel features used to build container. From a System Security perspective how may that be problematic?

Container-based sandbox

- Share kernel but separate user space resources
- Performance wise more efficient
- … but concept of container do not really exist in the kernel
  - Use combination of features: namespaces, overlayfs etc…
  - Give the illusion of being “alone on the system”
- Attack surface is much larger than in full virtualization
- Why?

Container-based sandbox

- Share kernel but separate user space resources
- Performance wise more efficient
- … but concept of container do not really exist in the kernel
  - Use combination of features: namespaces, overlayfs etc…
  - Give the illusion of being “alone on the system”
- Attack surface is much larger than in full virtualization
- Why?

Homework/exam question:
Compare and contrast the security of VMs and Containers

University of BRISTOL

What else could we do?

Unikernel

bristol.ac.uk

Unikernel

- VMs are quite heavy
  - A whole OS stack for a single httpd server
- Containers are not very strongly isolated
  - Information can leak through hardware in VMs case
  - … now we have a whole “buggy” OS to worry about

Unikernel

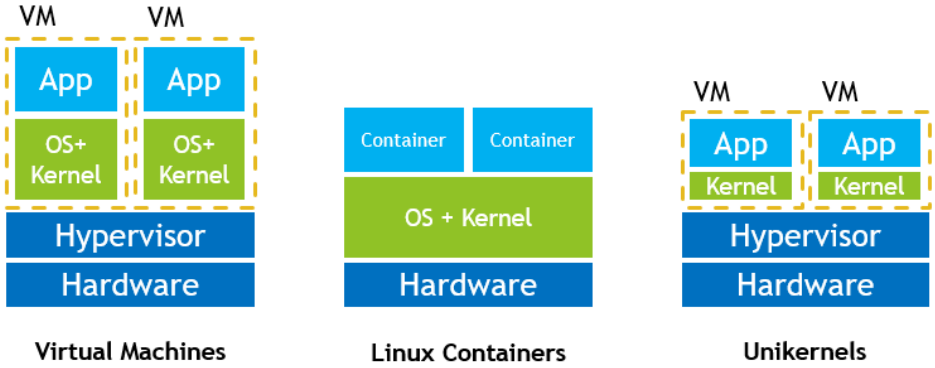
- VMs are quite heavy
  - A whole OS stack for a single httpd server
- Containers are not very strongly isolated
  - Information can leak through hardware in VMs case
  - … now we have a whole “buggy” OS to worry about
- Could I run my application on top of an Hypervisor?

Unikernel

- VMs are quite heavy
  - A whole OS stack for a single httpd server
- Containers are not very strongly isolated
  - Information can leak through hardware in VMs case
  - … now we have a whole “buggy” OS to worry about
- Could I run my application on top of an Hypervisor?
  - Yes you can
  - It is called a unikernel!

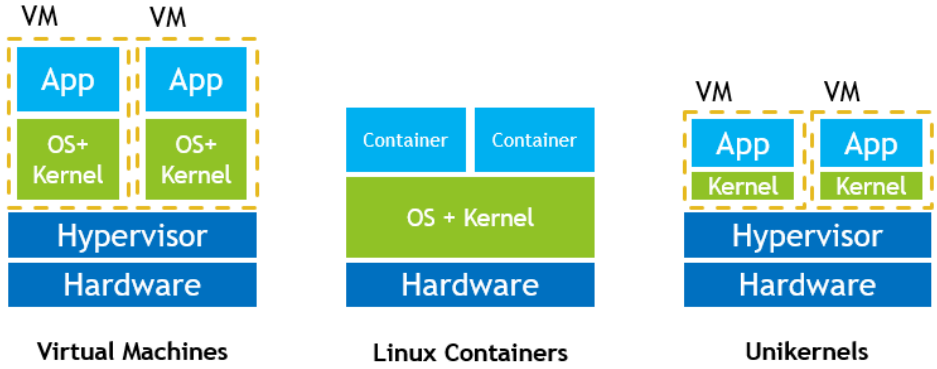
Unikernel

- VMs size a few MB



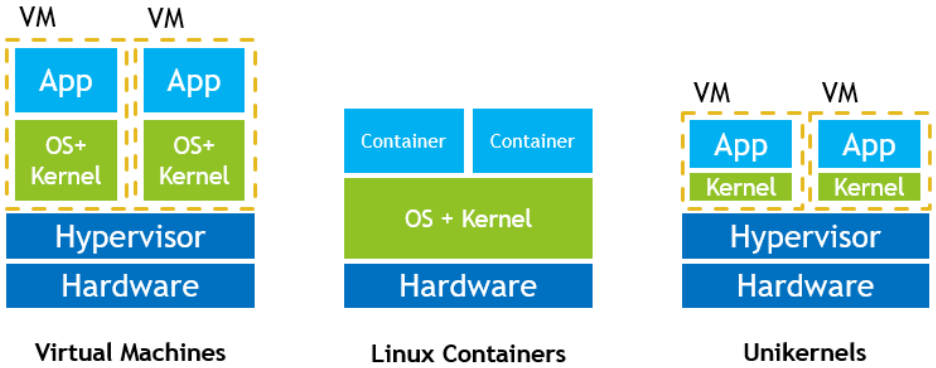
Unikernel

- VMs size a few MB
- Check paper: “Unikernels: Library Operating Systems for the Cloud”



Unikernel

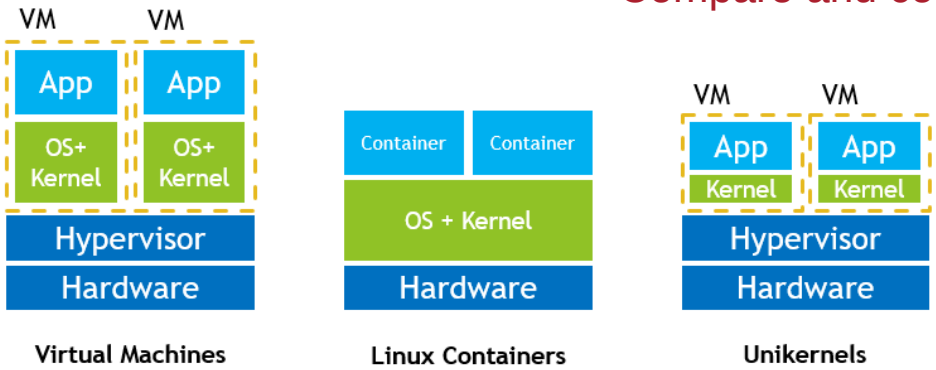
- VMs size a few MB
- Check paper: “Unikernels: Library Operating Systems for the Cloud”
- VMs vulnerabilities are still there



Unikernel

- VMs size a few MB
- Check paper: “Unikernels: Library Operating Systems for the Cloud”
- VMs vulnerabilities are still there

Homework/exam question:
Compare and contrast security



SGX Hardware supported enclave

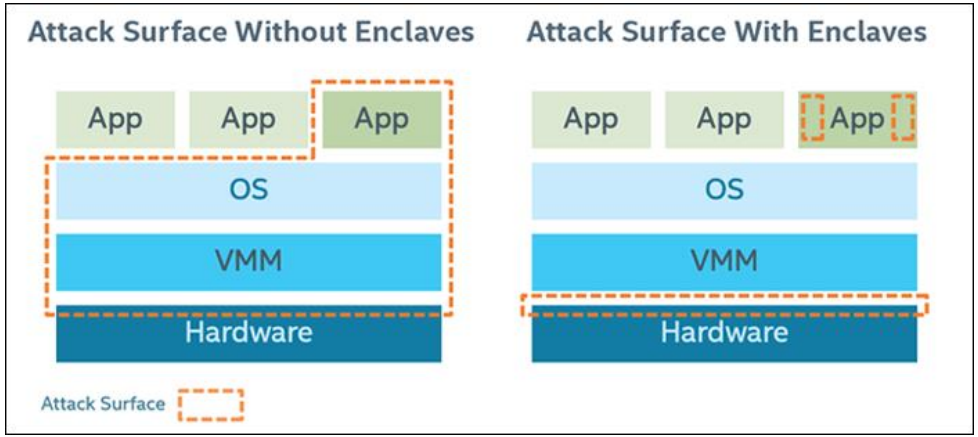
- WARNING: there is vulnerability in SGX

SGX Hardware supported enclave

- WARNING: there is vulnerability in SGX
- Idea: run an application within some isolation unit so it cannot be affected by the OS
  - don't trust the OS or the VMM/hypervisor
  - only need to trust the hardware

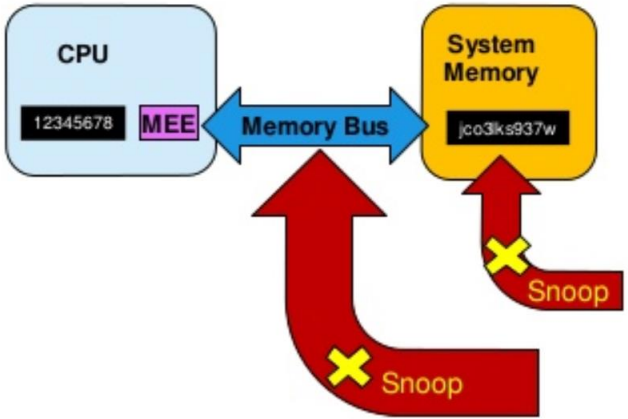
SGX Hardware supported enclave

- WARNING: there is vulnerability in SGX
- Idea: run an application within some isolation unit so it cannot be affected by the OS
  - don't trust the OS or the VMM/hypervisor
  - only need to trust the hardware
  - reduce attack surface

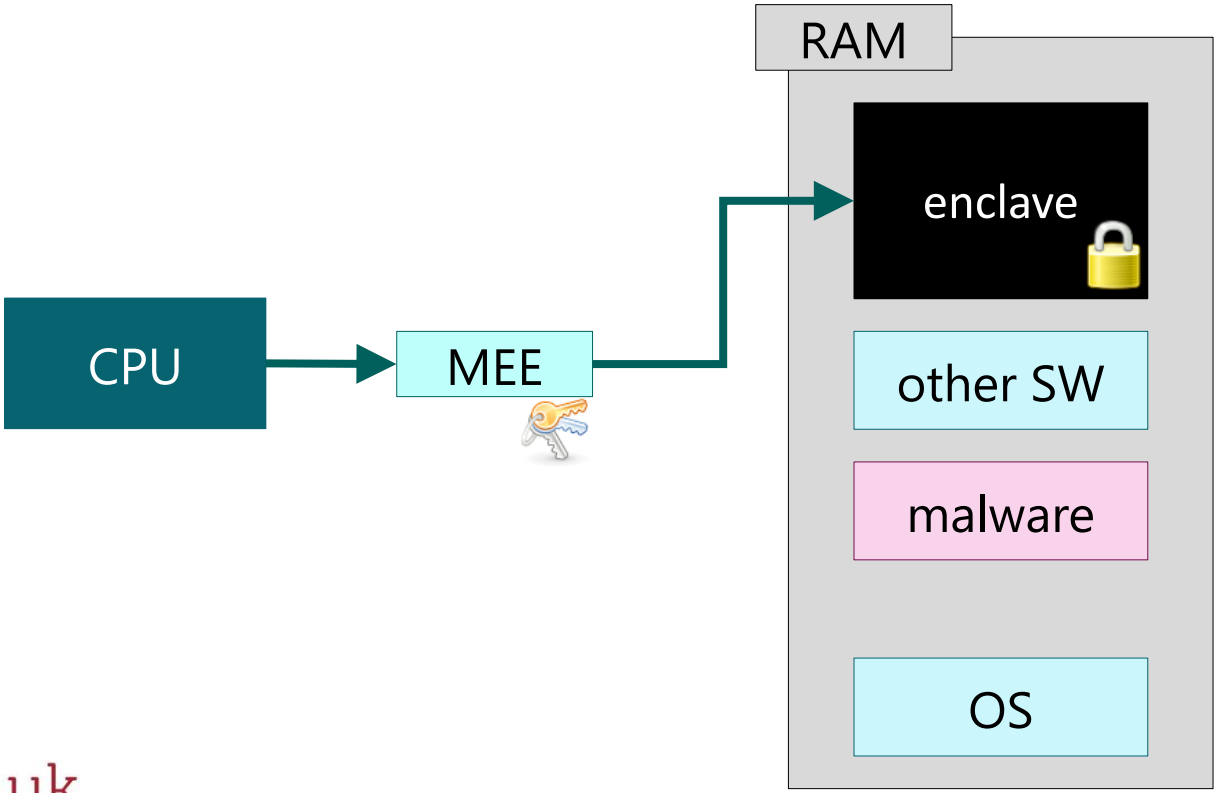


SGX: prevent memory snooping attack

- In CPU data is unencrypted
- Data outside CPU is encrypted
- External memory read/snooping only see encrypted data

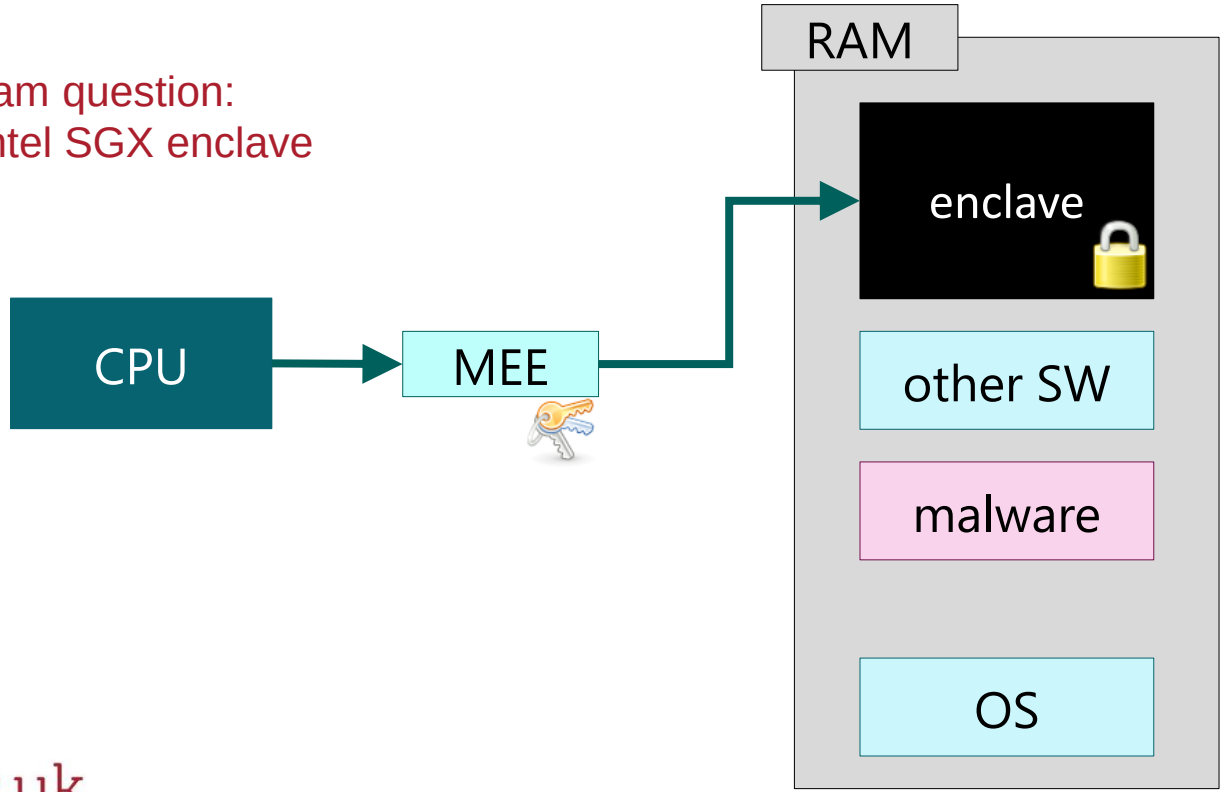


Memory protection



Memory protection

Homework/exam question:
Explain how Intel SGX enclave work.



University of BRISTOL

Problem?

bristol.ac.uk

University of BRISTOL

Problem?

High switching cost

bristol.ac.uk

Plan

- Need for isolation
- Virtual Machines
- Containers
- Unikernels
- SGX Enclaves

University of BRISTOL

Thank you, questions?

Office MVB 3.26

bristol.ac.uk