Hardware Enclaves & Intel SGX

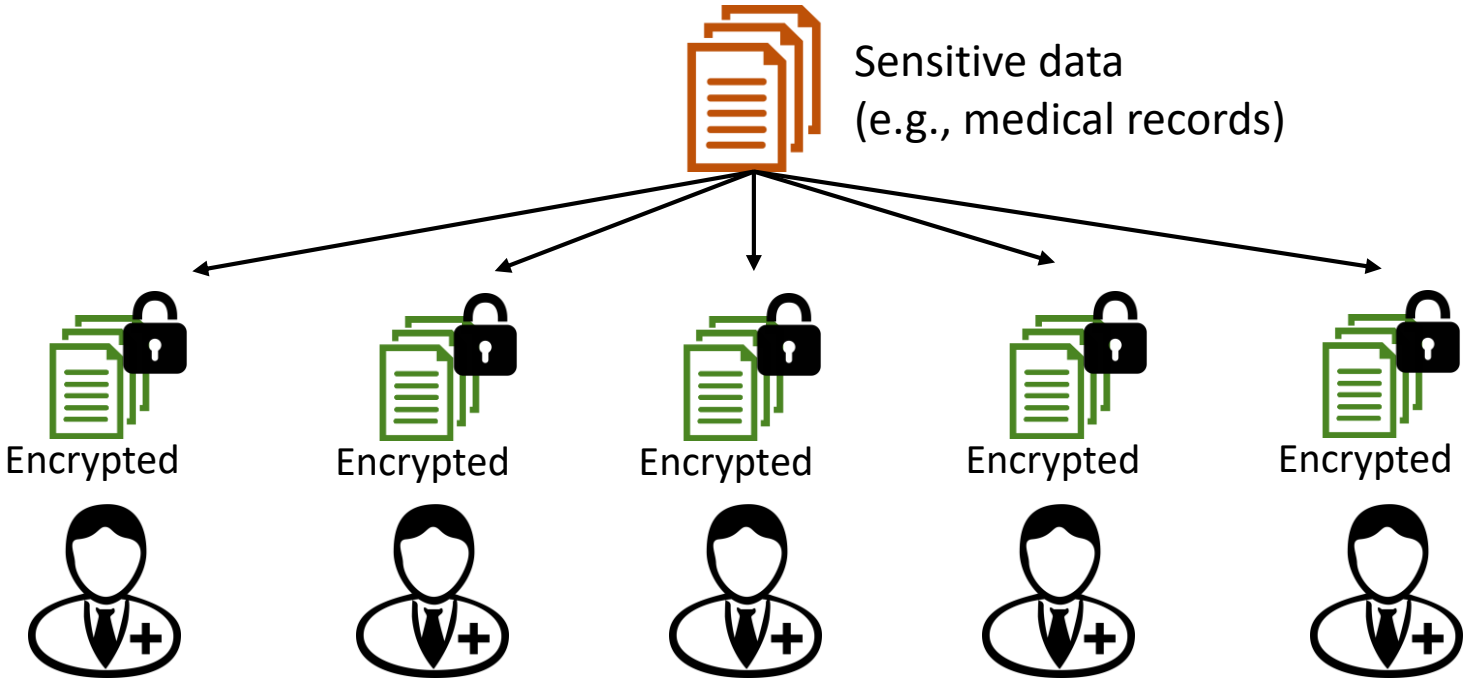
CS261

Hardware Enclaves

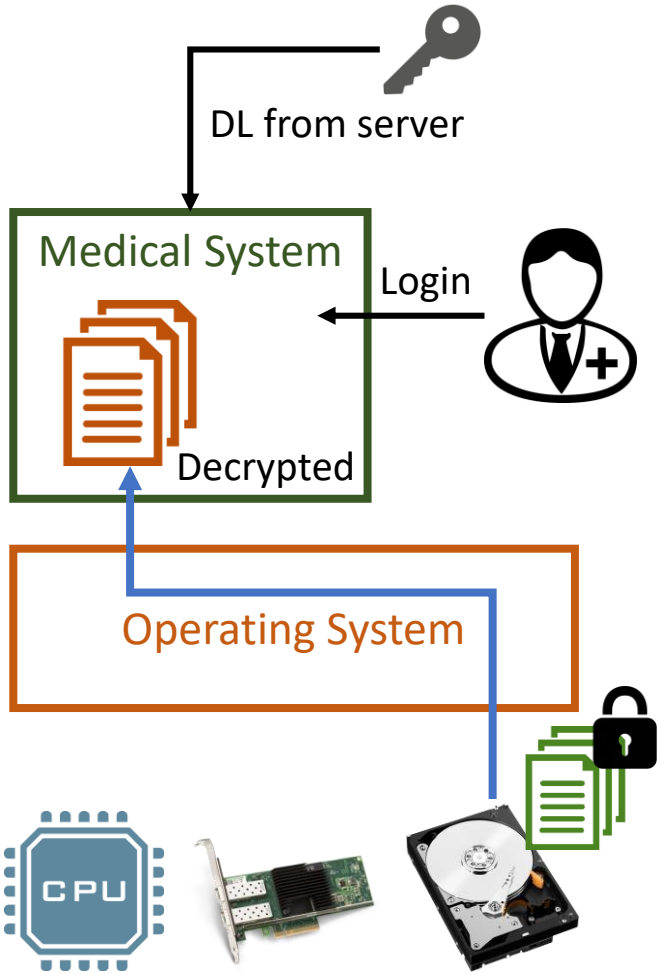
- HW abstractions for distributing trusted execution to untrusted platforms

Hardware Enclaves

- HW abstractions for distributing trusted execution to untrusted platforms

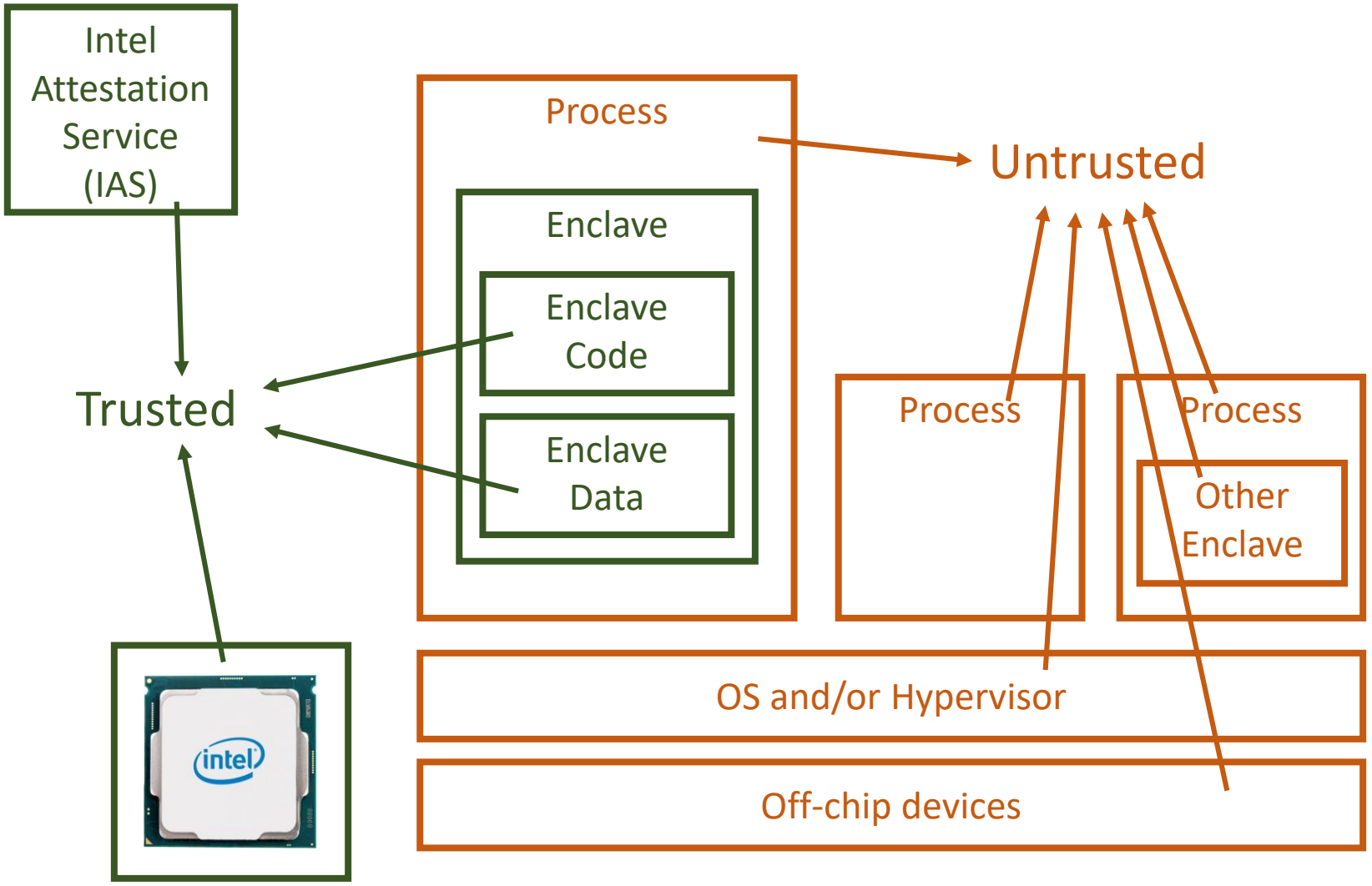


System Threats to Trusted Execution



- What can go wrong?
  - Side channels
    - out of scope for Intel SGX
  - Counterfeit software
  - Inject rootkits into OS
  - Privilege escalation
  - Install malicious kernel
  - Compromised HW devices
  - Cold-boot attacks

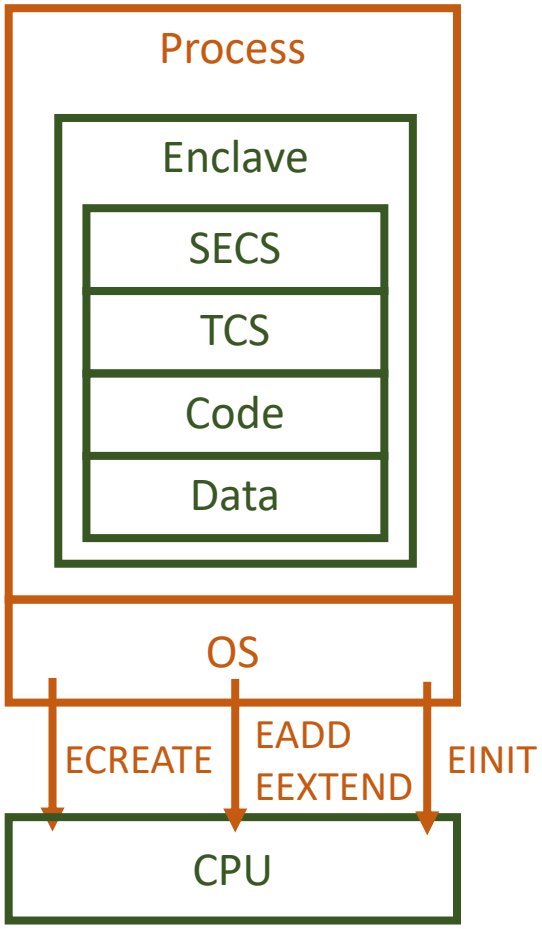
Threat Model of Hardware Enclaves



Elements of Hardware Enclaves

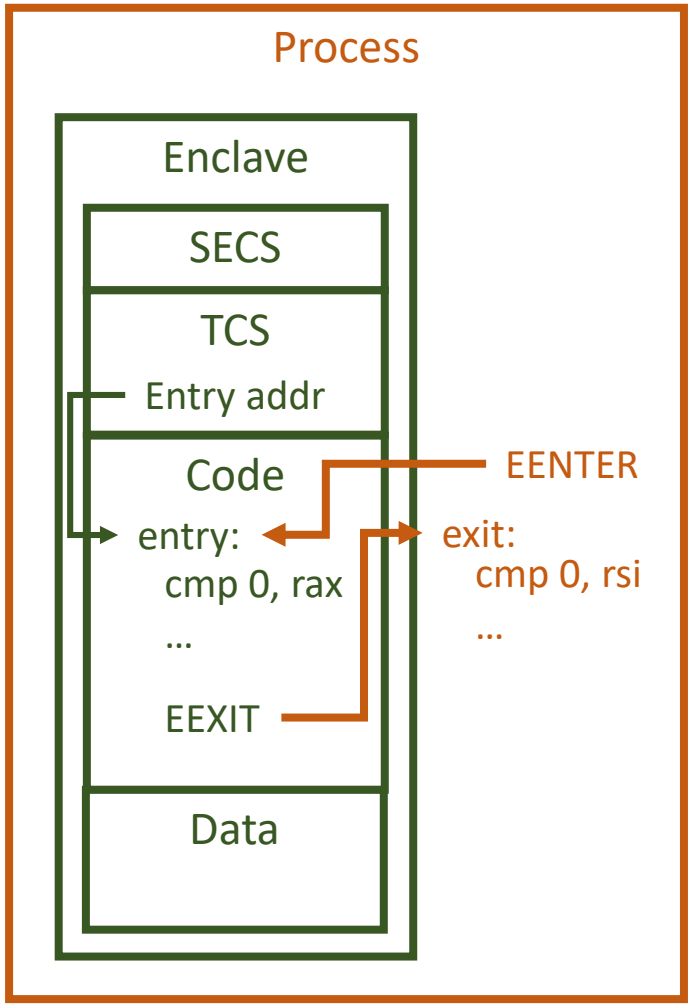
- Secure boot: HW-verified measurement + first instruction
- On-chip program isolation
- Cryptographically protected external memory
- Execution integrity; no interference from attackers
- Attestation and/or secret sealing

Enclave Creation with Intel SGX



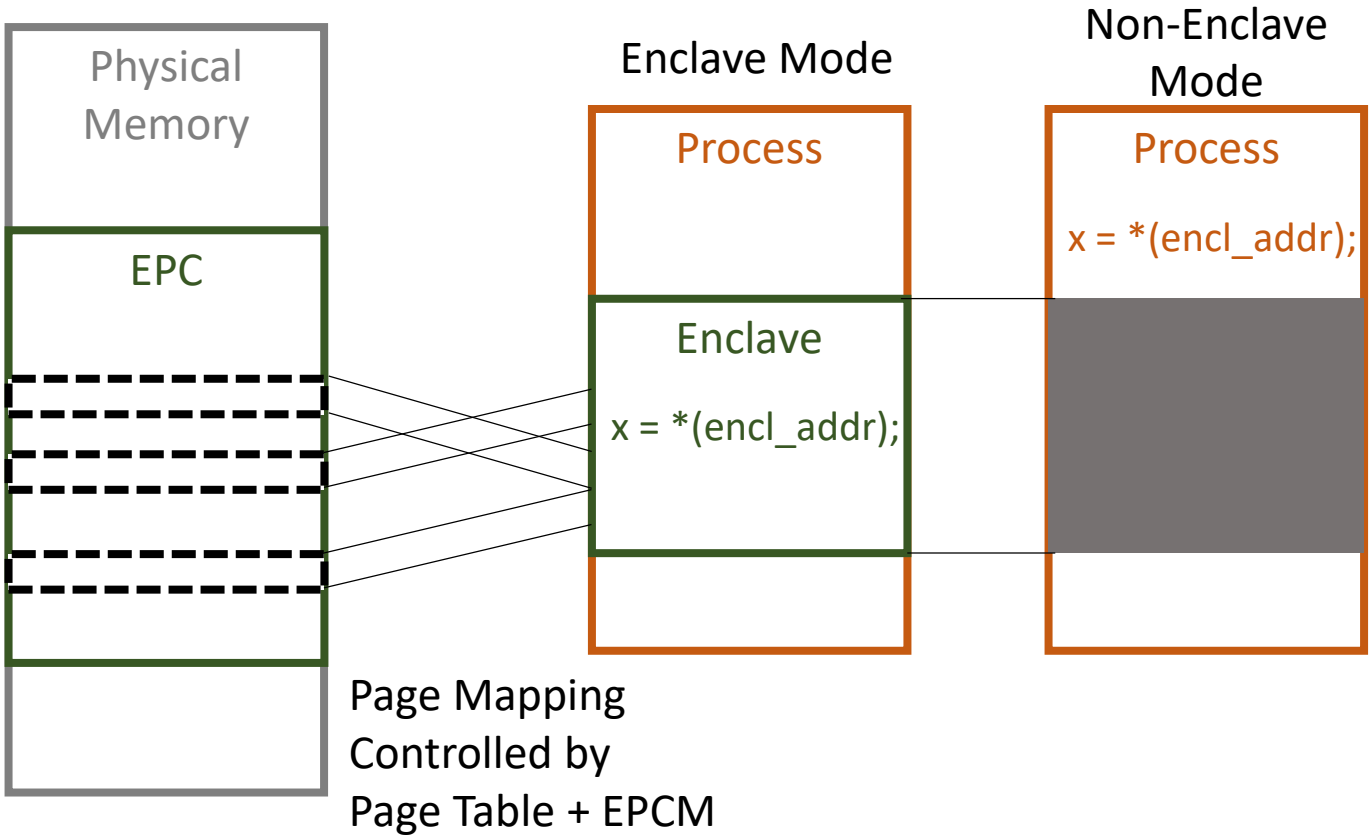
- ECREATE(SECS):
  create an enclave range
- EADD(SECS, addr, prot),
  EEXTEND(SECS, addr):
  add a page to enclave and measure the content
- EINIT(SECS, license):
  check & initialize an enclave

Enclave Enter & Exit



- EENTER(SECS, TCS):
  enter at a static enclave addr
- EEXIT(addr):
  exit enclave to any addr
- Enclave can accept parameters after the entry
- Attackers cannot interfere control flow unpredictably

Enclave Isolation



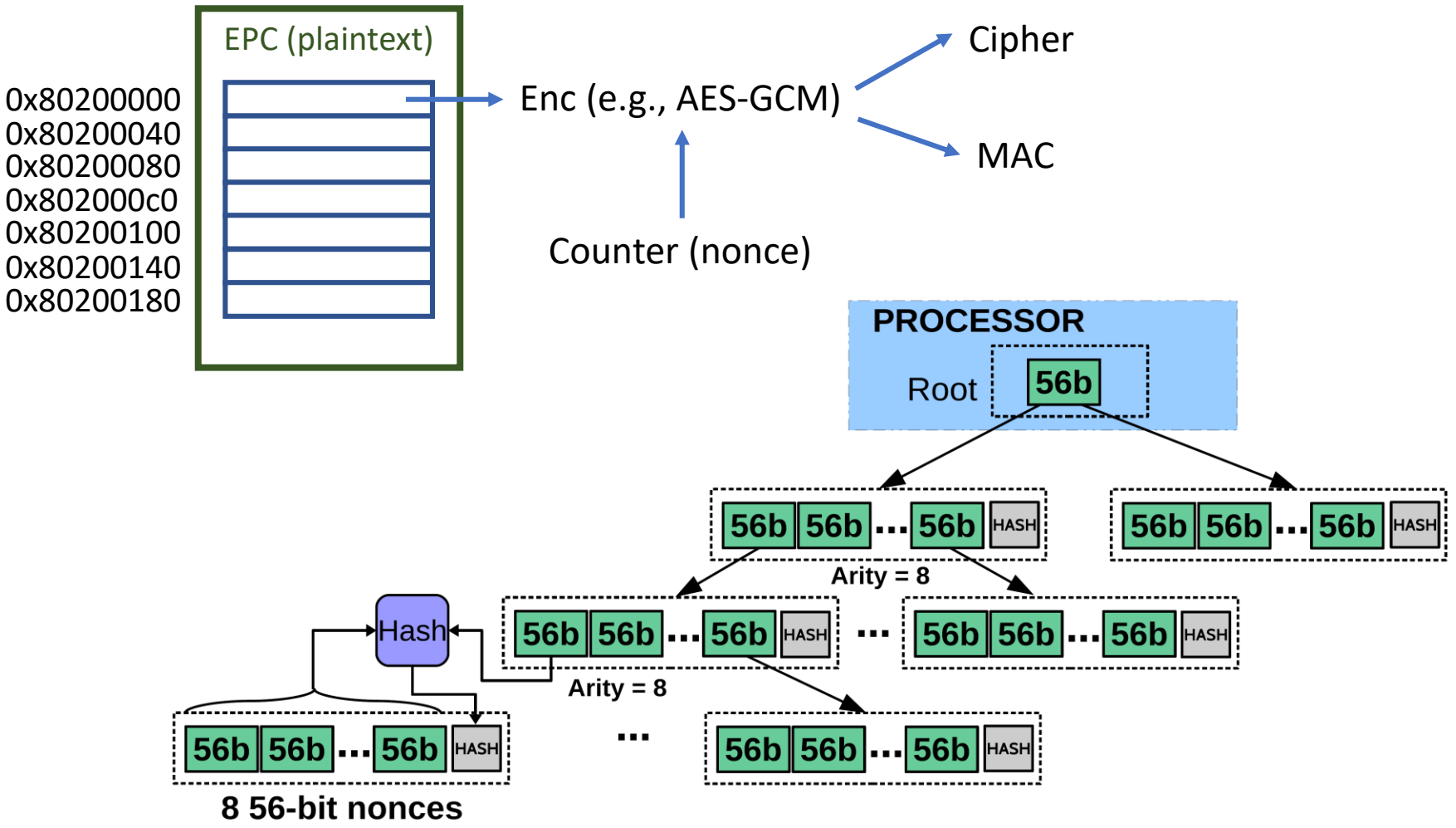
Abort page semantic:

EPC pages contains all 0s for execution outside the enclave

Memory Encryption Engine

- EPC pages are encrypted in DRAM
- Memory Encryption Engine (MEE) sits at the edge of CPU, connected to Memory Controller (MC)
- Cachelines are decrypted at cache misses, and re-encrypted when being written back to DRAM

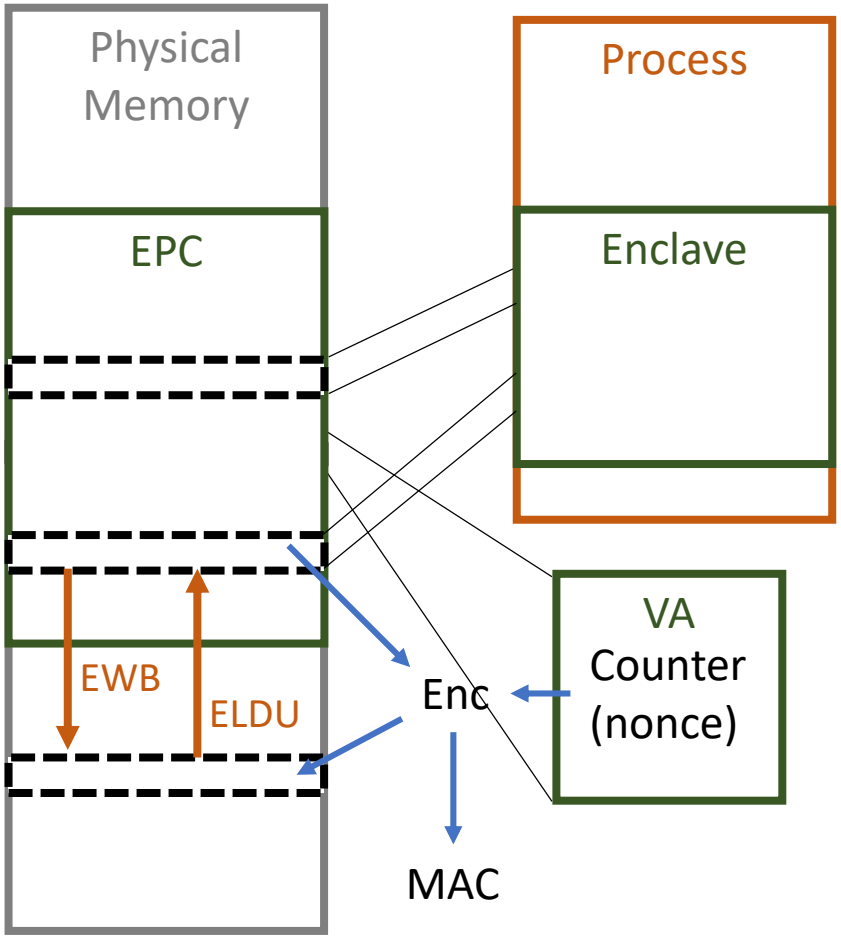
Memory Encryption Engine



EPC Paging

- EPC pages are limited: currently 93.5 MB on each platform
- Untrusted OS swaps the pages for enclaves
- Swapped-out pages are not in EPC, so no longer protected by MEE

EPC Paging



- EWB:
  copy a EPC page to non-EPC page
- ELDU:
  copy a non-EPC page to EPC page

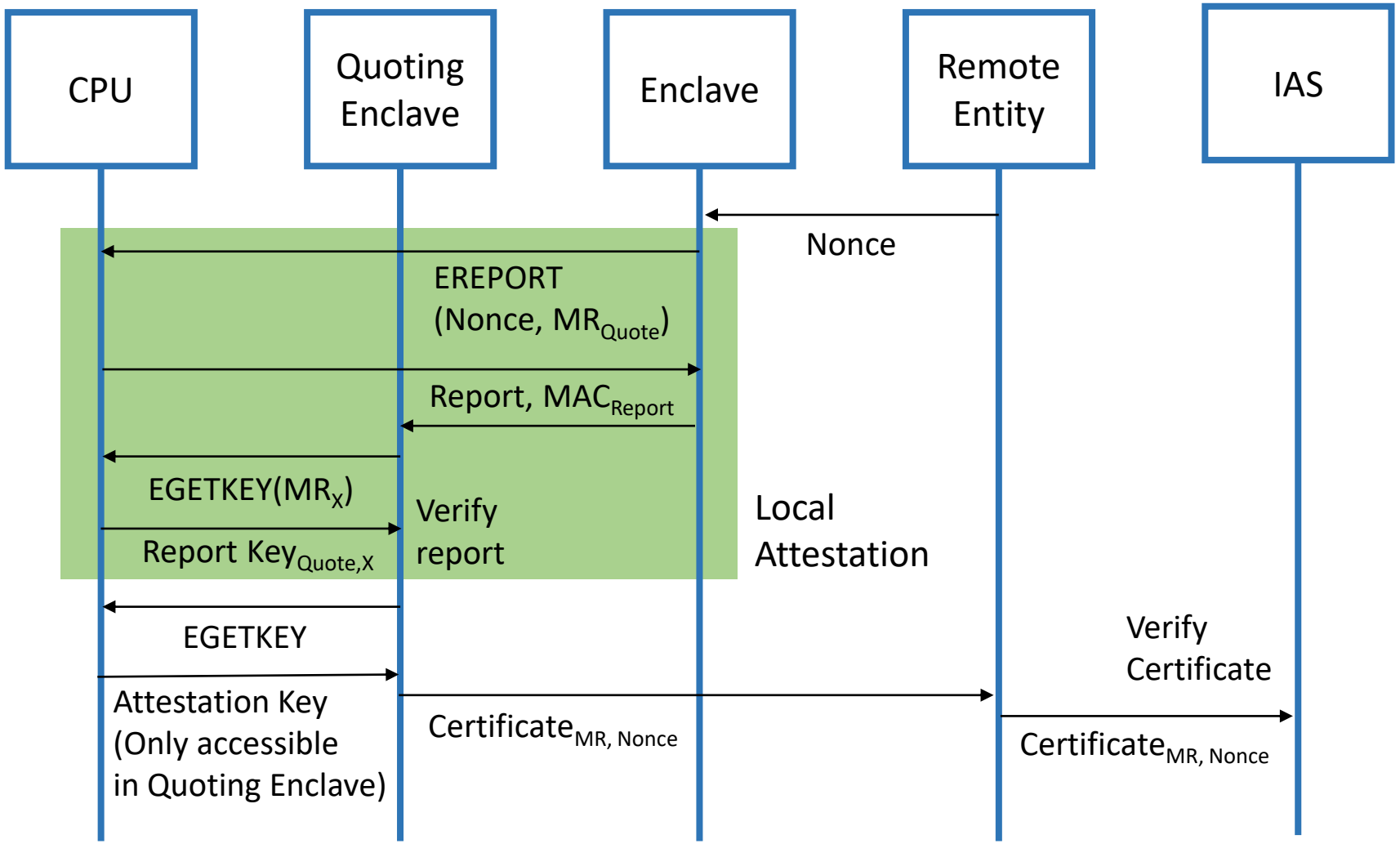
Execution Integrity

- Program states in either enclave memory or registers
- Enclave can be interrupted
  - Page faults (Paging)
  - Scheduling events
  - Exceptions or signals
- Interrupt → Asynchronous Exit (AEX)
  - Register values dumped inside enclave before exit
  - OS can only: (1) resume the enclave execution
    (2) re-enter enclave for exception handling

Attestation

- Proof that the program runs in a genuine enclave
- Each enclave has a set of unique keys
  - Report key – intra-platform (local) attestation
  - Attestation key – inter-platform (remote) attestation
  - Seal key – Sealing enclave secrets
  - Other keys – see Intel SDM
- Generated by a root secret (EPID) hidden in Intel CPU
  - Verified by Intel Attestation Service

Attestation Procedure



Use Cases for Hardware Enclaves

- Digital Right Management (DRM)
- Computation outsourcing, NFV
- Distributed system, edge computing, blockchains
- Alternative to HME or MPC
- Protection for antivirus, JIT compilers, etc
- Used for concealing attacks

Questions?

Hardware Enclaves & Intel SGX