Confidential Computing

UK RISE Annual Conference 2020

Ilhan Gurel Ericsson 2020-11-30

New Cross-Industry Effort to Advance Computational Trust and Security for Ne Generation Cloud and Edge Computing

By The Linux Foundation | August 21, 2019

CONFIDENTIAL COMPUTING CONSORTIUM

Confidential Computing Membership Grows 60 Percent

Jun 30, 2020

The Confidential Computing Consortium, a Lin
community dedicated to defining and accelera
computing, has announced Accenture, AMD,
book, iExec, IoTeX, Mellanox, NVIDIA, Orac
fidential Computing Consortium to contribute

What Is Confidential C

Big tech companies are adopting a
model called confidential computin
while it'

MANAGE > SECURITY

Confidential Computing, the Next Big Thing Making Cloud Scary for Enterprises

Why Confidential Computing Is a Game Changer

Confidential Computing is a transformational technology that should be part of every enterprise cloud deployment. It's time to start unlocking the possibilities together.

Confidential Computing Will Revolutionize The Internet Of Things

Computing Platforms & Problem Statement

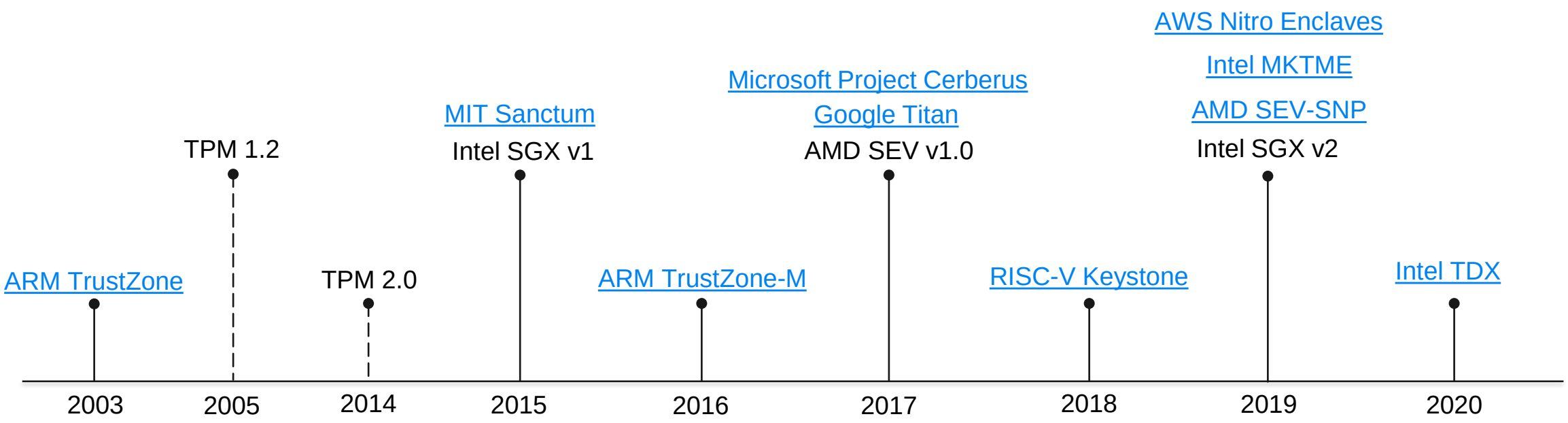
- Compromise of confidentiality
  - code and data are exposed in plain text on computing platforms
    - when "in use"
    - compromise of user data leading to the loss of privacy
  - from constrained IoT devices to cloud deployments
- Compromise of code and data integrity
- How to ensure that computing platforms are trustworthy and correct software is run on them?
- Making HW Root of Trust (RoT) available to guests in cloud deployments is challenging
- Compromise of IPR
  - e.g. algorithms, ML models,..

What does Confidential Computing mean?

- **Code and data confidentiality** can be achieved at **runtime ("in use")**
  - e.g. by means of **HW based isolation** and **memory/CPU state encryption technologies**
- **Data confidentiality and integrity** can be protected **at rest**, **in use** and **in transit**
- **Code and data cannot be tampered and accessed from outside of the trust boundaries of a secure enclave/trust domain**
- Code and data can be **measured** and **attested.**
  - **Confidential Computing is built upon the existing concepts of Trusted Computing**

... and the relevant technologies allow all of these to be achieved both on **bare metal** and in **virtualized environments**

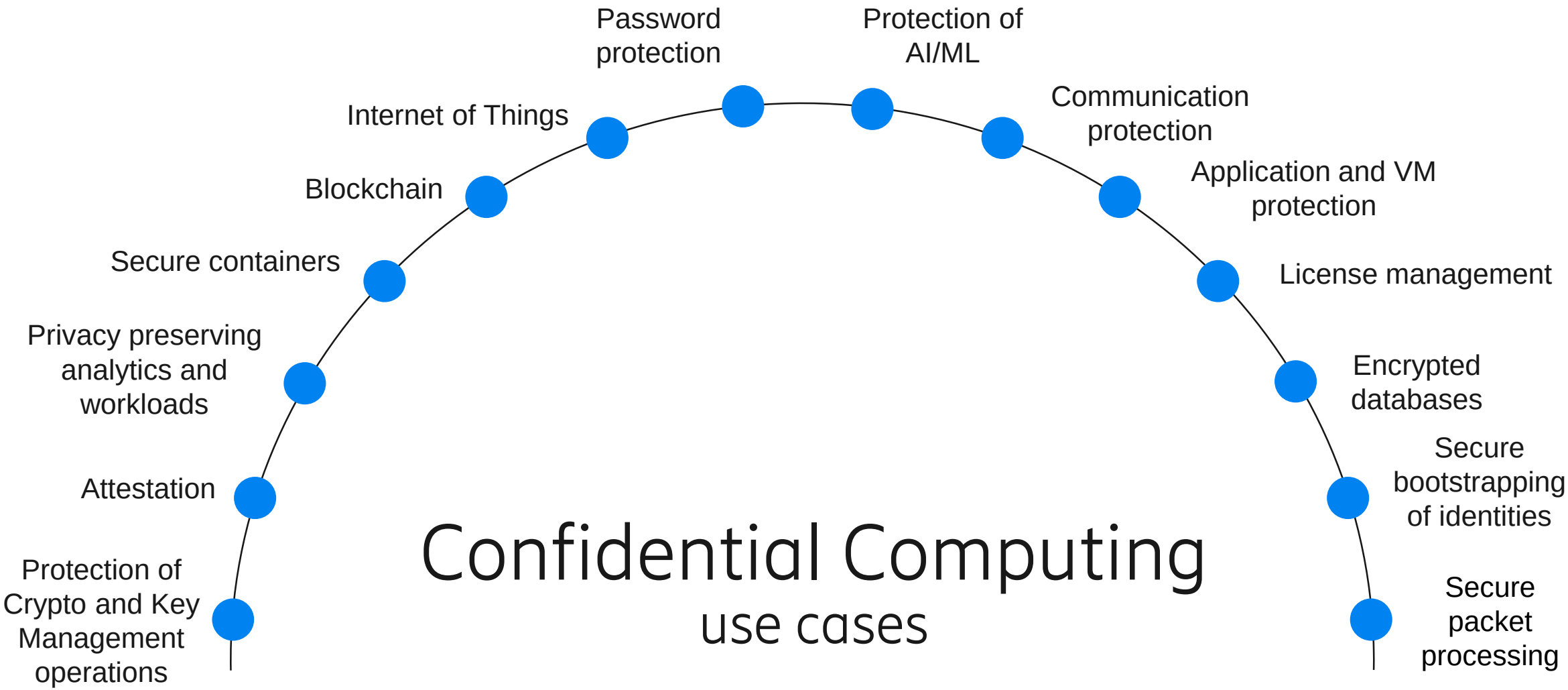
The relevant technologies



Intel MKTME (Multi Key Total Memory Encryption)
Intel TDX (Trust Domain Extensions)
Intel SGX (Software Guard Extensions)

AMD SEV (Secure Encrypted Virtualization)
AMD SEV-SNP (Secure Encrypted Virtualization – Secure Nested Paging)

The use cases



The relevant open source projects

- Intel SGX SDK
  - provided by Intel for developing Intel SGX secure enclaves
  - provides libraries, headers, samples codes, tools and documentation
- Open Enclave SDK
  - open source SDK that mostly hides underlying enclave technology and initiated by Microsoft
- Google Asylo
  - open source SDK that mostly hides underlying enclave technology and initiated by Google
- Google Project Oak: Control and end to end encryption of data in distributed systems
- Apache Teaclave
- Crypto API Toolkit (Intel SGX based softHSM)
- Baidu Rust SGX SDK
- Fortanix Rust SGX SDK
- Enarx
- RISC-V Keystone
- RISC-V HexFive
- Hyperledger Private Data Objects (Blockchain by Intel)
- The Confidential Consortium (CoCo) Framework (Blockchain by Microsoft)
- Hyperledger Fabric Private Chaincode (Blockchain by IBM)

What else is happening?

- The Confidential Computing Consortium
  - Announced on August 21st, 2019 by the Linux Foundation
  - An industry wide effort
    - to advance computational trust and security for next-generation computing
    - to bring together HW and SW vendors, cloud providers, developers, open source experts and academics to accelerate the confidential computing market
    - to influence the relevant technical and regulatory standards
- Growing number startup companies emerging and providing services/products related to Confidential Computing
- Commercial availability of "Confidential Computing" capabilities by the cloud vendors
- Lots of academic research on
  - finding novel solutions utilizing Confidential Computing Technologies
  - microarchitectural side channel attacks and mitigations
  - some interesting projects such as Slalom, Project Graviton and CoSMIX

The solutions for encrypting applications, containers and VMs

OCI encrypted container images

- Allows encryption container later
- Encryption can be done by using Containerd imgcrypt library or skopeo tool
- Also see OCIcrypt and the specification proposal. The work also includes Kubernetes integration

Enarx

- An application deployment framework
- Support both AMD SEV an Intel SGX. Intel TDX to follow
- In case of AMD SEV, Enarx allows deployment of encrypted workloads to AMD SEV after attestation and key provisioning processes

Intel SGX Protected Code Launch (PCL)

- PCL allows running encrypted code and data
- "sgx_encrypt" tool encrypts the sections of a secure enclave (except .bss, .tbss, .dynamic, .debug,..) by using AES GCM
- Content key is provisioned y using a sealing enclave IP enclave by using `sgx_create_encrypted_enclave()`

VMware vSphere VM encryption

- VMware vSphere VM encryption allows encryption of VM images and VM disk images
- Integration with vCenter Server and KMS

What about Homomorphic Encryption?

- Homomorphic Encryption (HE) refers to an encryption scheme "that allows computation to be directly on encrypted data, without requiring any decryption in the process"
- Invented in 2009
  - but the origins go back to a paper (titled as "On Data Banks and Privacy Homomorphisms") published by Ronald Linn Rivest and Len Adleman in 1978
  - *(Ronald L. Rivest invented RSA algorithm together with Adi Shamir and Len Adleman in 1977)*
  - the existence of a Full HE scheme was demonstrated in 2009 by Craig Gentry
- Publicly available SW implementations are available: Microsoft SEAL , HELib (IBM) and PALISADE
- FHE is far from being practical due to massive overhead in computation and memory

Attestation is a process of measuring code and data; and reporting these measurements as digitally signed to a requesting entity, which can evaluate these measurements further according to known values or whitelists.

What is needed for a fully functional attestation mechanism ?

HW RoT

HW RoT for storage, reporting and measurement

Crypto functionality

Cryptographic identities for attestation

Attester

SW APIs for accessing crypto modules/the secure enclaves functionality

Attestation agent and the relevant services

Protocol

An attestation protocol between attestor and verifier

Efficient and scalable protocol that can mitigate known attacks (e.g. replay attacks)

Verifier

Being able to validate, verify and evaluate attestation reports

Technical capabilities for updating whitelists, etc.

Keeping whitelists up to date

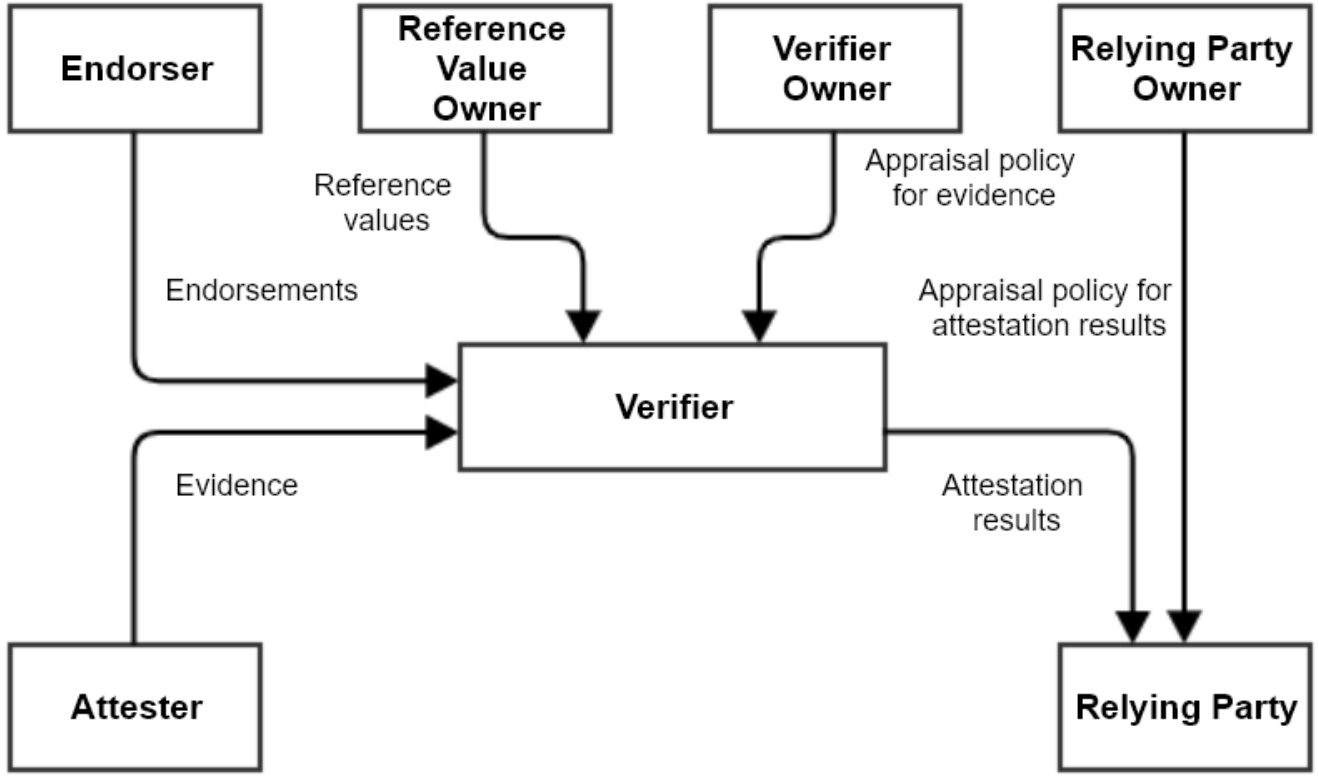
Relying Party

Relies on attestation verification results provided by the Verifier

Applies specific actions based on attestation results

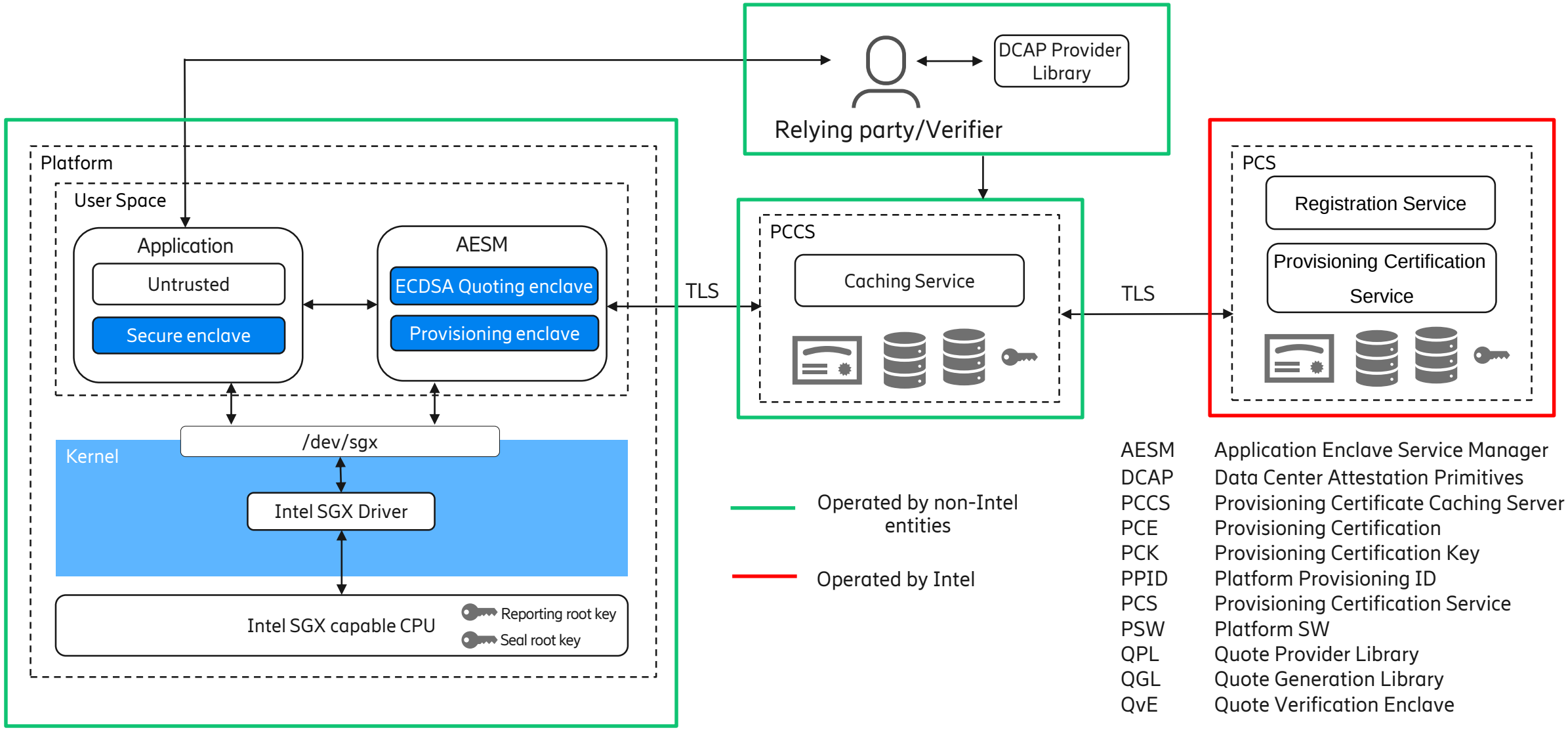
IETF RATS (Remote ATtestation procedureS) architectural overview

Appraisal Policy for Evidence	Appraisal Policy for Evidence: A set of rules that informs how a Verifier evaluates the validity of information about an Attester
Attester	An entity providing evidence that must be appraised in order to infer the extent to which the Attester is considered trustworthy
Endorser	An entity (typically a manufacturer) whose Endorsements help Verifiers appraise the authenticity of Evidence
Evidence	A set of information (digitally signed) about an Attester that is to be appraised by a Verifier
Relying Party	A role performed by an entity that depends on the validity of information about an Attester, for purposes of reliably applying application specific actions
Relying Party Owner	An entity (typically an administrator), that is authorized to configure Appraisal Policy for Attestation Results in a Relying Party
Verifier	A role performed by an entity that appraises the validity of Evidence about an Attester and produces Attestation Results to be used by a Relying Party



https://datatracker.ietf.org/doc/draft-ietf-rats-architecture/

Intel SGX DCAP overview



AESM	Application Enclave Service Manager
DCAP	Data Center Attestation Primitives
PCCS	Provisioning Certificate Caching Server
PCE	Provisioning Certification
PCK	Provisioning Certification Key
PPID	Platform Provisioning ID
PCS	Provisioning Certification Service
PSW	Platform SW
QPL	Quote Provider Library
QGL	Quote Generation Library
QvE	Quote Verification Enclave

Intel SGX DCAP attestation data

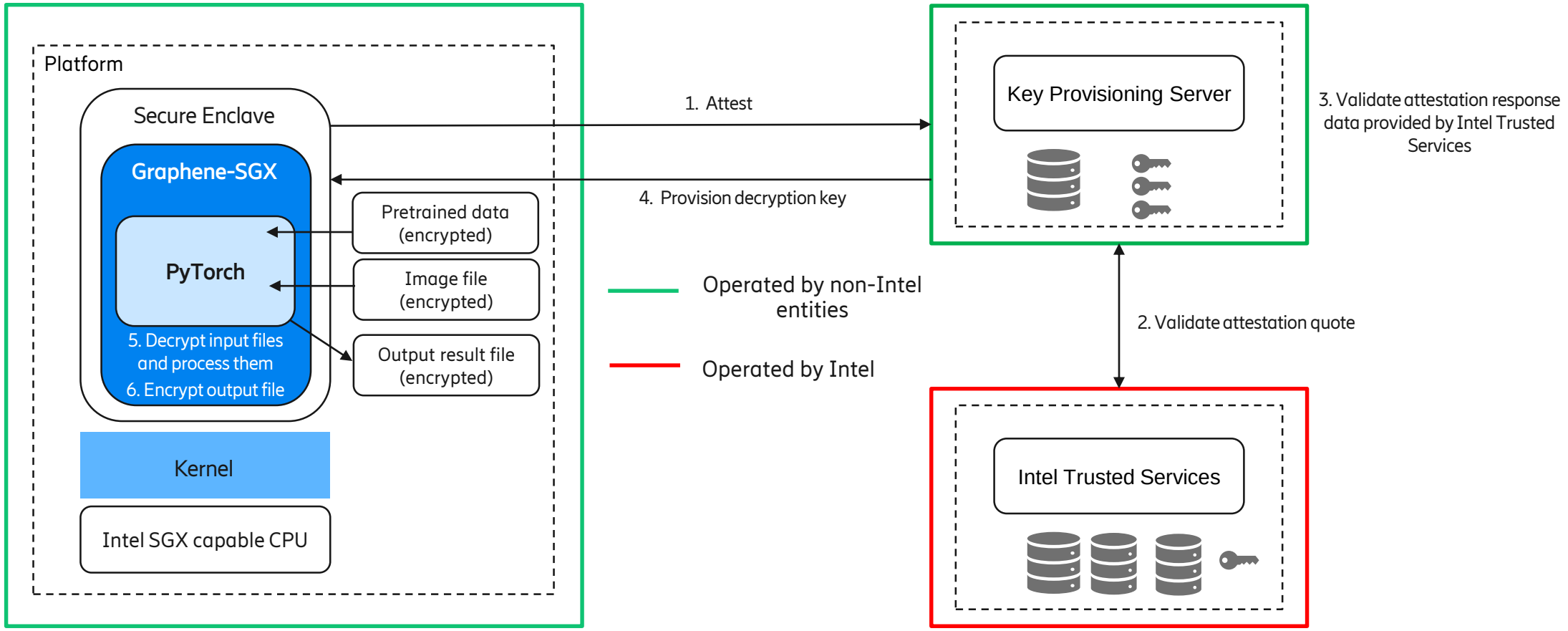
- Attestation report body includes the following information (see `sgx_report_body_t`)

```
typedef struct _report_body_t
{
    sgx_cpu_svn_t           cpu_svn;        /* (  0) Security Version of the CPU */
    sgx_misc_select_t       misc_select;    /* ( 16) Which fields defined in SSA.MISC */
    uint8_t                 reserved1[SGX_REPORT_BODY_RESERVED1_BYTES];  /* ( 20) */
    sgx_isvext_prod_id_t    isv_ext_prod_id;/* ( 32) ISV assigned Extended Product ID */
    sgx_attributes_t        attributes;     /* ( 48) Any special Capabilities the Enclave possess */
    sgx_measurement_t       mr_enclave;     /* ( 64) The value of the enclave's ENCLAVE measurement */
    uint8_t                 reserved2[SGX_REPORT_BODY_RESERVED2_BYTES];  /* ( 96) */
    sgx_measurement_t       mr_signer;      /* (128) The value of the enclave's SIGNER measurement */
    uint8_t                 reserved3[SGX_REPORT_BODY_RESERVED3_BYTES];  /* (160) */
    sgx_config_id_t         config_id;      /* (192) CONFIGID */
    sgx_prod_id_t           isv_prod_id;    /* (256) Product ID of the Enclave */
    sgx_isv_svn_t           isv_svn;        /* (258) Security Version of the Enclave */
    sgx_config_svn_t        config_svn;     /* (260) CONFIGSVN */
    uint8_t                 reserved4[SGX_REPORT_BODY_RESERVED4_BYTES];  /* (262) */
    sgx_isvfamily_id_t      isv_family_id;  /* (304) ISV assigned Family ID */
    sgx_report_data_t       report_data;    /* (320) Data provided by the user */
} sgx_report_body_t;
```

- Intel SGX DCAP attestation quote. (see `sgx_quote3_t`)

```
typedef struct _sgx_quote3_t {
    sgx_quote_header_t  header;
    sgx_report_body_t   report_body;
    uint32_t            signature_data_len;
#ifdef _MSC_VER
#pragma warning(push)
#pragma warning ( disable:4200 )
#endif
    uint8_t             signature_data[];
#ifdef _MSC_VER
#pragma warning(pop)
#endif
} sgx_quote3_t;
```

Demo: PyTorch running in a secure enclave with encrypted input and output files



Graphene-SGX: https://github.com/oscarlab/graphene/

https://arxiv.org/pdf/2009.04390.pdf

https://www.ericsson.com/en/security

Are the secure enclave technologies secure?

There is no binary “yes/no” answer to this question and the answer depends on:

Adversaries and their capabilities

- adversaries with the advanced technical capabilities such as being able to initiate powerful attacks

Deployments

- with or without physical access to devices?
- secure key generation and provisioning?

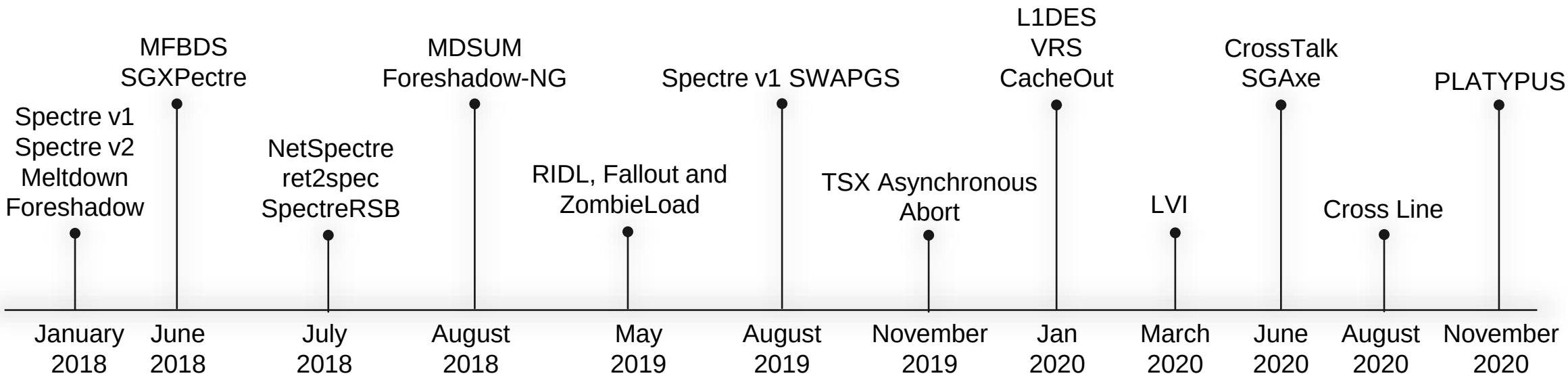
HW & SW security vulnerabilities

- security vulnerabilities?
- are the known vulnerabilities patched?

Supply chain security

- both HW and SW supply chain security
- well established vulnerability and incident management processes?

Timeline of the microarchitectural side channel vulnerabilities



L1DES	L1D Eviction Sampling
MFBDS	Microarchitectural Fill Buffer Data Sampling
MDSUM	Microarchitectural Data Sampling Uncacheable Memory
VRS	Vector Register Sampling
LVI	Load Value Injection
MLPDS	Microarchitectural Load Port Data Sampling
RIDL	Rogue In-flight Data Load
TSX	Transactional Synchronization Extensions