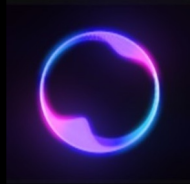
5. ProvisioningとTCB Recovery

Ao Sakurai

2023年度セキュリティキャンプ全国大会
L5 - TEEの活用と攻撃実践ゼミ

本セクションの目標



- 後のセクションで解説するEPID方式の**リモートアテステーション**（Remote Attestation; 以降**RA**と省略）で使用する**秘密情報をマシンにデプロイ**する**プロビジョニング**という処理を解説する

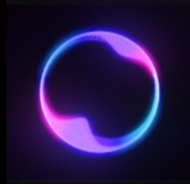
- 危殆化したTCBを安全な状態に回復するための**TCB Recovery**について簡単に解説する

Provisioning（プロビジョニング）

用語集

用語	説明
Attestationキー	主にRemote Attestationにおいて非常に重要な役割を果たす鍵で、Provisioningにより生成される。 その実はEPIDメンバ秘密鍵。
Provisioning Enclave (PvE)	Provisioning処理において中核的な役割を果たす。 Architectural Enclaveの一つ。
Quoting Enclave（QE）	PvEが生成しストアしたAttestationキーをロードし、RAにおいてQUOTE構造体を作成する。 Architectural Enclaveの一つ。
Intel Provisioning Service (IPS)	Provisioning処理において、PvEとやり取りを行うIntel側のサービス。
Intel Key Generation Facility (iKGF)	RPK（後述）やEPID関連鍵等、様々な鍵を作成しストアする、Intelの鍵作成管理施設。 インターネットからは接続できない場所に隔離され、強固に保護されている。

Intel EPID（Enhanced Privacy ID）

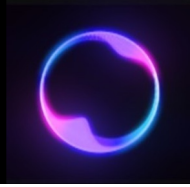


- ある**（1つの）グループ**に対し**複数のメンバ**を**匿名の状態**で対応させる事が出来るスキーム

- **直接匿名認証**（Direct Anonymous Attestation; **DAA**）の**応用的な実装例**である

- EPIDの**メンバ秘密鍵**で署名すると、EPIDの**グループ公開鍵**を用いて、**署名者を特定する事なく検証**できる
  - 例：あるマシンが特定のCPUグループに属しているかを匿名のまま検証

Provisioningの概要（1/2）

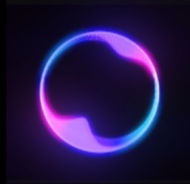


- SGXマシンが**正当なCPUやSWを搭載**しているかを確認した後、ある**EPIDグループのメンバとして加入**させ、**Attestationキーを獲得**させる処理
  - Intel CPUにおけるEPIDグループは、CPUの種類（Core i3, i5, i7）と、セキュリティバージョン番号（SVN）が同一であるような**数百万個のCPUをカバー**している

- Provisioning処理は、SGXマシン側では**Architectural Enclave**（AE）の1つである**Provisioning Enclave（PvE）**が中心となる

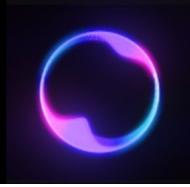
- Intel側は**Intel Provisioning Service（IPS）**が中心となる

Provisioningの概要（2/2）



- Provisioning処理は、その**マシンの初使用時**の他、購入後にファームウェア、BIOS、マイクロコード等の**重要なシステムコンポネントが更新**された際にも実行される（**TCB Recovery**）

Provisioningに使用する鍵一覧（1/2）

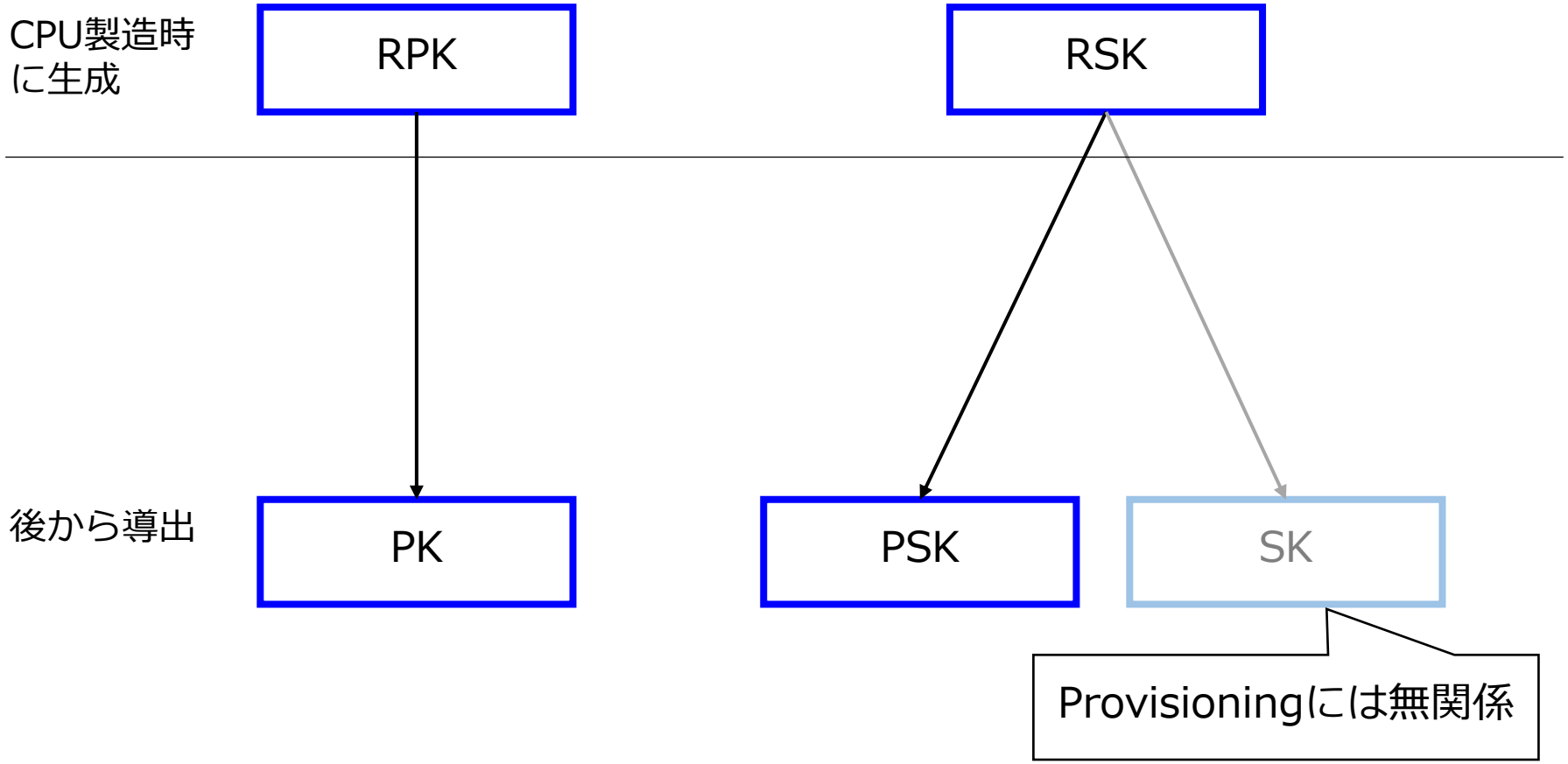


- **Intelによる命名**が**非常に紛らわしい**が、以下のように整理する事が出来る

鍵名	概要
Root Provisioning Key（RPK; またはProvisioning Secret）	CPU製造時に各CPUのe-fuseに焼き付けられる秘密情報。**この値はIntel側もiKGFで管理・保持している**。
Root Seal Key（RSK; またはSeal Secret）	CPU製造時に各CPU内で乱数的に生成され、e-fuseに格納される秘密情報。**この値はIntel側も保持・把握していない**。
Provisioning Key（PK）	RPKから導出されるプロビジョニング鍵。Provisioningの手続きで使用される。
Provisioning Seal Key（PSK）	RSKから導出される、Provisioning手続き上で必要なシーリングを行う為のシーリング鍵。

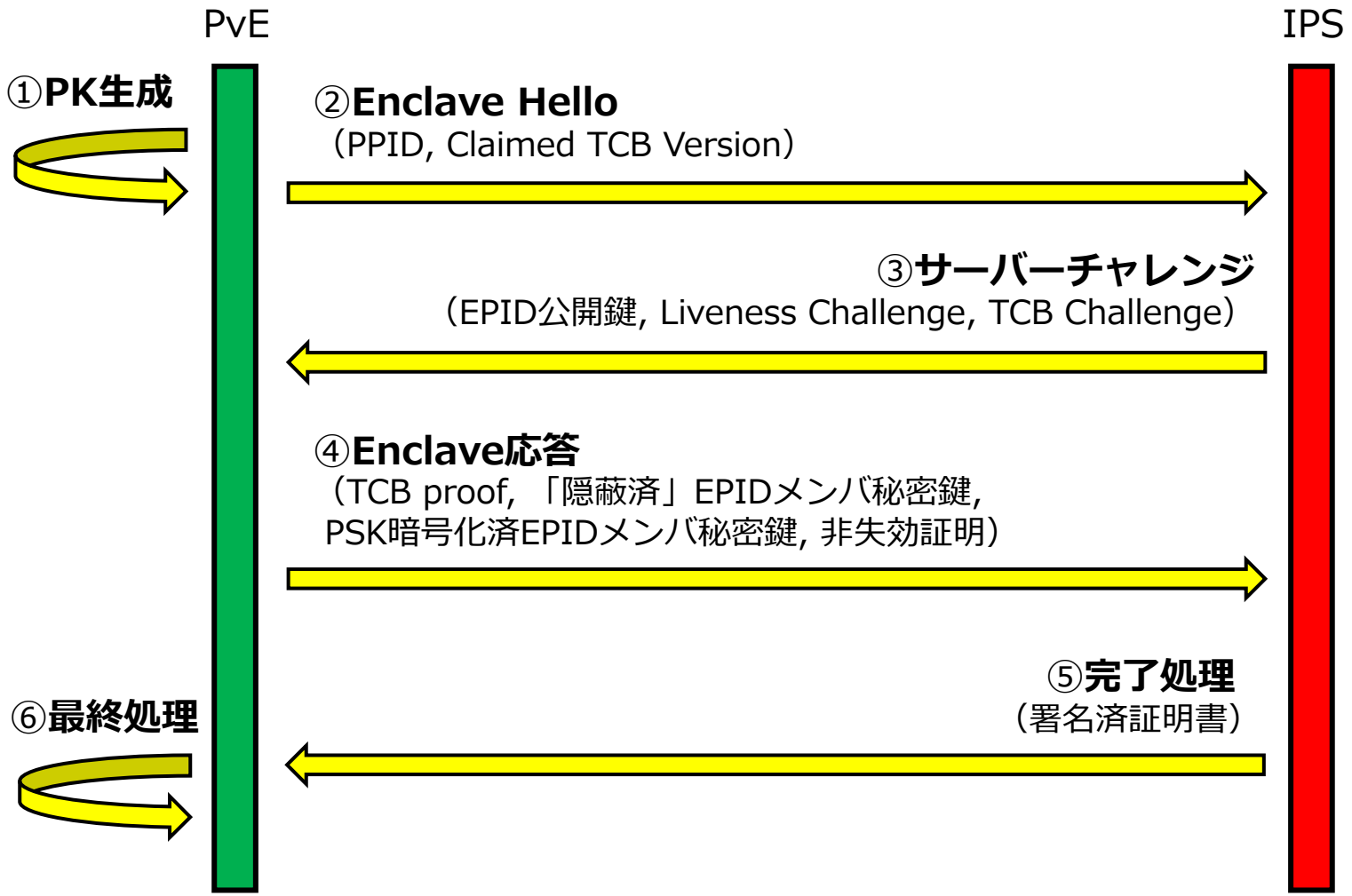
（参考）Seal Key：通常のシーリングに使用される鍵。これもRSKからポリシ（MRENCLAVE、MRSIGNER）に応じて生成されるが、PSKとは違いOWNEREPOCHという値を有する等の違いがある。

Provisioningに使用する鍵一覧（2/2）



Provisioningフロー

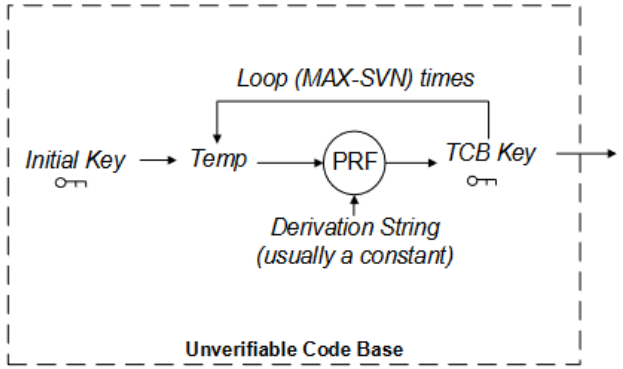
- Provisioningフローを図示すると以下のようになる：



PKの作成

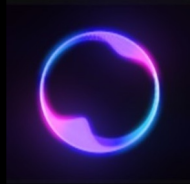
- 簡単に言えば、RPKからHW・SWそれぞれの特定のコンテキストを付与してPKを導出する

RPKのHW-TCBとのバインド：
プロセッサ内で生成されるInitial KeyをTCBのSVN（Security Version Number）回だけPRF（擬似ランダム関数）に通し、TCB鍵を生成する。一次情報に記載は無いが、Initial KeyはRPKであると思われる。



SW情報の付与：
EGETKEY命令を発行し、上記TCB鍵をベースとして各種SW情報（例：CPUSVN、MRSIGNER）を付与し、これをPKとする

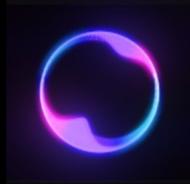
Enclave Hello



- PvEは、以下2つの値を生成する。
  - **PPID**：PKのハッシュ値
  - **Claimed TCB version**：TCBSVNから導出される値

- 上記2つの値を生成後、**IPSの公開鍵**で双方ともに暗号化し、Enclave HelloとしてIPSに送信する。
  - この「**IPSの公開鍵**」が何物なのかは**文献からでは解読不能**だが、気合で探した所**以下のソースコードにハードコーディング**されている事が判明：
    https://github.com/intel/linux-sgx/blob/1efe23c20e37f868498f8287921eedfbcecdc216/psw/ae/data/constants/linux/peksk_pub.hh

サーバーチャレンジ



- IPSは、PvEから送信された**PPIDで検索を実行**し、そのPvEの
  プラットフォームが**以前にProvisioningされた事があるか**を**確認**する

- 確認結果に応じて以下のように**サーバーチャレンジ**をPvEに返送する：

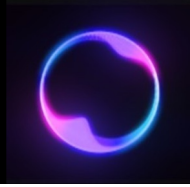
Provisioningが初であった場合
PvEのプラットフォームを**加入させるEPIDグループを決定**後、以下の情報を
PvEに返信する：

- EPIDグループパラメータ（EPID公開鍵）
- Liveness Challenge
- iKGFで予め作成済みのTCBチャレンジ

Provisioning履歴があった場合

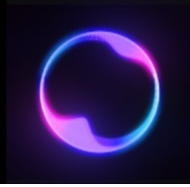
上記に加え、以前生成したAttestationキーをPSKで暗号化したものを
TCBチャレンジに追加し返信する

Enclave応答（1/5）



- サーバーチャレンジをIPSから受け取ったら、PvEは自身の**プラットフォームの正当性**を証明するために、**以下の4つの値**を準備する：
  - TCB proof
  - 「数学的に隠蔽」されたEPIDメンバ秘密鍵
  - PSKによって暗号化済のEPIDメンバ秘密鍵
  - 非失効証明

Enclave応答（2/5）- TCB proof

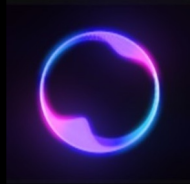


- 以下の2通りの方法のいずれかで、TCBが正当である事を証明する（**TCB proof**の作成）。sgx101のサイトでは後者のみを記載している

方法①
PvEは、iKGFでPKを用いて作成された暗号化済乱数（Liveness Challange）を、自身の持つPKで復号する。
ちなみに、**iKGFが全てのRPKを有している**事から、IPSは**各PKを容易に再現して生成可能**であるらしい（詳細は不明）。

方法②
PvEは、受信したTCBチャレンジをPKで復号し、その復号済TCBチャレンジを鍵として、Liveness ChallengeのCMACを生成する。

Enclave応答（3/5）- EPIDメンバ秘密鍵

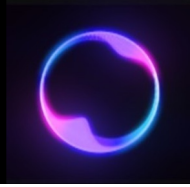


- その後、PvEは**EPIDメンバ秘密鍵**（＝**Attestationキー**）を作成し、EPIDのプロトコルに従って「数学的に隠蔽」する
  - 「数学的に隠蔽」する方法の詳細は不明だが、**何らかの群論的な操作**をするのではないかと産総研でのレクチャ時に候補として上がった

- それとは別に、このEPIDメンバ秘密鍵を**PSKで暗号化**する

- このEPIDメンバ秘密鍵は**全てPvE内で生成処理が完結する**ため、**Intel側がこの値を知る事はできない**
  - Intel SGX Explained[5]では**あたかもIPSがこの鍵を送信している**かのような図を載せているが、**これは誤り**

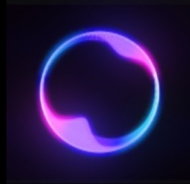
Enclave応答（4/5）- 非失効証明



- 過去にProvisioningを行った事がある（=**再Provisioningである**）場合、そのPvEのプラットフォームが過去に一度も**失効**（Revoke）していない事を**証明**する必要がある

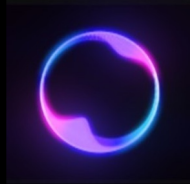
- 具体的には、サーバから取得したバックアップ済Attestationキーのコピーを PSKで復号し、それを使ってIntelが選択したメッセージに署名する事で証明する

Enclave応答（5/5）



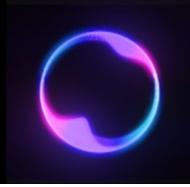
- PvEは、作成した以下のデータをIPSに送信する。
  - TCB proof
  - 「数学的に隠蔽」されたEPIDメンバ秘密鍵
  - PSKによって暗号化済のEPIDメンバ秘密鍵
  - 非失効証明

完了処理



- IPSはPvEからのEnclave応答中の**TCB proof**を、iKGFから取得した値を用いて**検証**する。
- 検証の結果成功である場合、続いて対象PvEのプラットフォームを**EPIDグループに加入**させる。
- IPSは、応答中の「数学的に隠蔽された」EPIDメンバ秘密鍵と、IPSの持つEPIDグループ発行者鍵を用いて、**署名済証明書**を作成する。
- IPSは、上記で作成した署名済証明書を同梱したメッセージをPvEに返送する（Provisioningの完了）。

最終処理

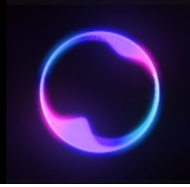


- 「数学的に隠蔽された」EPIDメンバ秘密鍵と、PSK暗号化済メンバ秘密鍵は、今後の再Provisioningの為IPS側に保存される
  - 非失効証明の作成時に参照されたバックアップ済Attestationキーは、まさに過去に保存されたこれである

- PvEは、AttestationキーをPSKでシーリングし、プラットフォーム上に保存する
  - RA時、Quoting Enclave（QE）はここから持ってくる
  - PSKは**OWNEREPOCH値を持たず**、かつこのシーリングは**MRSIGNERポリシ**である（PSKを使用した場合の仕様）為、**QEでも問題なくアンシーリング**出来る

TCB Recovery

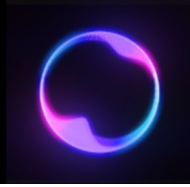
TCB Recovery



- **TCB（復習）**：SGXが安全に動作するために、正しく動作し、悪意や危殆化が存在しない事が求められるコンポネントの総称。
  CPU本体やマイクロコード、QE、PvE、SGXSDKのtRTSなど

- TCB Recoveryは、SGXのTCBが危殆化した際にCPUの**マイクロコード**や**SVN**を**アップデート**し、新規のAttestationキーを**再プロビジョニング**する処理
  - 新規のAttestationキーに対応するグループは**脆弱性対応済み**としてマークされているため、**既存の脆弱なグループと区別**され、RA時のマシンの安全性を判断する上での重要な材料となる

本セクションのまとめ



- 直接Enclave開発者の実装との関わりは薄いが、EPID方式のRAを行う上で重要な前提処理であるプロビジョニングについて学習した

- SGXのTCBが危殆化した際にTCBの安全性を回復する処理であるTCB Recoveryについて簡単に触れた

参考文献

[1] Intel® Software Guard Extensions:
EPID Provisioning and Attestation Services (https://cdrdv2.intel.com/v1/dl/getContent/671370)

[2] Attestation – SGX 101 (https://sgx101.gitbook.io/sgx101/sgx-bootstrap/attestation)

[3] Intel® Enhanced Privacy ID (EPID) Security Technology
(https://www.intel.com/content/www/us/en/developer/articles/technical/intel-enhanced-privacy-id-epid-security-technology.html)

[4] Intel® Software Guard Extensions Trusted Computing Base Recovery
(https://community.intel.com/legacyfs/online/drupal_files/managed/01/7b/Intel-SGX-Trusted-Computing-Base-Recovery.pdf)

[5]”Intel SGX Explained”, Victor Costan & Srinivas Devadas, https://eprint.iacr.org/2016/086.pdf

[6]” Intel® Software Guard Extensions Trusted Computing Base Recovery”, Intel,
https://www.intel.com/content/www/us/en/developer/articles/technical/software-security-guidance/resources/intel-sgx-software-and-tcb-recovery-guidance.html