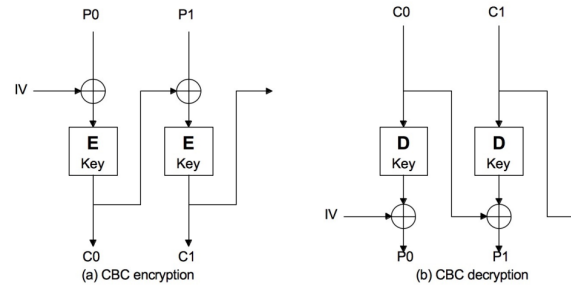


# 1 TPM

## 1.1 Chiffrements

### 1.1.1 symétrique

- Une seule clé pour crypter et décrypter
- Codage par bloc ou par bloc chaîné



- `openssl enc -aes-256-cbc -e -in t.txt -out t.enc //encrypt`
- `openssl enc -aes-256-cbc -d -in t.enc -out t.txt//decrypt`

### 1.1.2 asymétrique

- Deux clés (publique et privée) clé publique disponible par des certificats (CMD pgp)
- Encrypt public -> Decrypt private => confidentialité
- Encrypt private-> Decrypt public => signature digitale

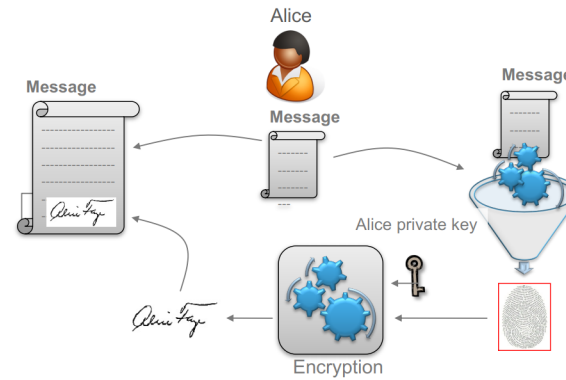
### 1.1.3 hash

Transforme un texte, document en un nombre de N bits unique (SHA-2, SHA-3, Blake2).

`md5sum file => a6a0e8d0522e2c5de921b1c455506320` où `openssl dgst -md5 file MD5(file)= a6a0e8d0522e2c5de921b1c455506320`

### 1.1.4 signature

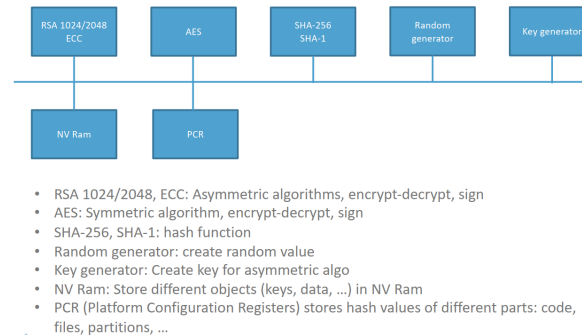
En deux parties: 1. Calcul du HASH puis encryptage avec clé privée.



## 1.2 Implémentations TPM

- discrete : Circuit dédié
- integrated : Partie du  $\mu C$  qui gère le TPM
- Hypervisor : virtuel fournis par personne fiable
- Software : virtuel pour faire des test pas sécurisé

## 1.3 Architecture interne



## 1.4 Hiérarchies

- endorsement : réservé au fabricant du TPM et fixé lors de la fabrication.
- platform : réservé au fabricant de l'hôte et peut être modifier par l'équipementier.
- owner : hiérarchie dédiée à l'utilisateur primaire du TPM peut être modifié en tout temps.
- null : réservé aux clés temporaires (RAM s'efface à chaque redémarrage)

## 1.5 Créer, utiliser clés

```

Create RSA endorsement key: tpm2_createprimary -C e -G rsa2048 -c e_primary.ctx
Create RSA platform key:    tpm2_createprimary -C p -G rsa2048 -c p_primary.ctx
Create RSA owner key:       tpm2_createprimary -C o -G rsa2048 -c o_primary.ctx
Create RSA null key:        tpm2_createprimary -C n -G rsa2048 -c n_primary.ctx

```

## 1.6 Commandes principales

[style=bash]

*tpm2\_createprimary -Co -Grsa2048 -co\_prim créer un clé primaire ownertpm2\_getcaphandles -transient voir clé dans la RAM tpm2\_getcaphandles -persistent voir clé dans la NV -RAM tpm2\_evictcontrol -co\_primary.ctx sauve une clé en NV -RAM tpm2\_flushcontext! -te f f acertout la RAM tpm2\_create -Co\_prim -Grsa2048 -uchild\_pub -rchild\_priv créer clé en j Co\_prim - uchild\_pub - rchild\_priv - cchild charger clé en fant*

## 1.7 encrypter-décrypter, signer-vérifier

[style=bash]

*tpm2\_rsaencrypt -cchild -srsaes clear file -o encrypted f ile tpm2\_rsa decrypt -cchild -srsaes encrypted f ile -o clear f ile tpm2\_sign -cchild -gsha256 -o file.sign f ile tpm2\_verify s cchild - gsha256 - s file.sign - m file*