**Identifiants**

1. **sid** : l'identifiant de session est lié à l'identité des parties, aux paramètres mathématiques et à la clé publique.
2. **epid** : l'identifiant d'époque est lié à l'époque de rafraîchissement de la clé et au matériel de clé auxiliaire des parties pour cette époque (par exemple, les clés publiques Paillier).
3. **sgid** : l'identifiant de session de signature est lié au message en cours de signature, et deux sessions de signature différentes pour le même message sont distinguées par le sgid

**Threshold ECDSA** (ECDSA à seuil) : signature en ligne

1. Génération de Clé (Key Generation)
   1. input (keygen, sid, i)
      1. **keygen**: Cela indique que l'action à exécuter est la génération de clés.
      2. **sid** : Cela fait référence à un **identifiant de session**. Chaque session de génération de clés peut avoir un identifiant unique pour suivre le processus et éviter les conflits entre différentes sessions.
      3. **i** : Cela représente l'identifiant d'une partie ou d'un signataire dans le protocole. Dans les systèmes de signatures à seuil, plusieurs parties participent, et chaque partie a un identifiant unique.
   2. Phase de Génération de Clé : obtenir le partage de clé publique et la clé publique
   3. Phase d'auxiliaire : obtenir les paramètres supplémentaires nécessaires au processus.
   4. Sortie : inclut la clé publique (pub-key), l'identifiant de session (sid), et le partage de clé .
2. Signature
   1. **input** (sign, epid, sgid, i, msg)
      1. **sign**: Cela indique que l'action à exécuter est une signature.
      2. **epid**: L'identifiant de période epidepidepid est lié à une époque spécifique de rafraîchissement de clé dans le protocole.
      3. **sgid**: L'identifiant de session de signature sgidsgidsgid est associé spécifiquement à une instance de signature d'un message particulier.
      4. **i**: spécifie l'identifiant de la partie qui effectue l'opération de rafraîchissement de clé.
      5. **msg**: le message à signer
   2. **Phase de Pré-Signature** : Exécute la phase de pré-signature pour préparer les données nécessaires à la signature.
   3. **Hachage du Message** : Détermine le message à signer en appliquant une fonction de hachage, représentée par .
   4. **Obtention de la Signature** : Exécute la phase de signature en utilisant les données préalablement calculées pour obtenir la signature et une liste des parties corrompues (où si est une signature valide, sinon contiendra les identités des parties corrompues).
   5. **Sortie** : La sortie comprend la chaîne de signature, les identifiants, le message, et les valeurs de signature.
3. Rafraîchissement de Clé (Key Refresh)
   1. **input** : (key-refresh, sid, , i)
      1. **key-refresh** : Cela indique que l'action à exécuter est un rafraichissement de clé
      2. **sid** : Cela fait référence à un **identifiant de session**. Chaque session de génération de clés peut avoir un identifiant unique pour suivre le processus et éviter les conflits entre différentes sessions.
      3. : ensemble ordonné des clés publiques partagées par les participants au protocole.
      4. **i** : spécifie l'identifiant de la partie qui effectue l'opération de rafraîchissement de clé.
   2. **Phase auxiliaire** : Exécute la phase auxiliaire pour obtenir les nouvelles informations nécessaires au rafraîchissement.
   3. **Mise à Jour des Partages de Clé :** 
      1. Les parties effacent toutes les données de pré signature et de clé auxiliaire qui ne sont plus nécessaires.
      2. Réassigne l'identifiant de session et met à jour les partages de clé en fonction des nouvelles données.
   4. **Erreur** : En cas d'erreur lors du rafraîchissement de clé, toutes les futures invocations de ce processus doivent être ignorées. Cela évite les confusions ou les échecs supplémentaires dans la gestion des clés.

**Identifiant de Session (sid)**

**Description** : L'identifiant de session \( sid \) est un identifiant unique qui est associé à une instance spécifique du protocole de signature.

**Composantes Associées** (Identité des Parties) : Le \( sid \) permet d'identifier les parties qui participent à la session. Cela garantit que toutes les actions effectuées sous ce \( sid \) sont liées aux mêmes participants.

**Paramètres Mathématiques** : Les paramètres mathématiques nécessaires à l'exécution du protocole (comme le champ fini ou les générateurs de groupe) sont également associés à cet identifiant.

**Clé Publique** : \( sid \) est lié à la clé publique utilisée dans cette session. Cela peut inclure des clés qui sont générées ou partagées spécifiquement pour le contexte de cette session.

**Importance** : Le \( sid \) aide à éviter les collisions entre différentes sessions, garantissant que chaque session est distincte et que les données sont correctement associées à la bonne instance.

**Identifiant de Période (epid)**

**Description :** L'identifiant de période \( epid \) est lié à une époque spécifique de rafraîchissement de clé dans le protocole.

**Composantes Associées :** (Époque de Rafraîchissement de Clé) : \( epid \) fait référence à une période durant laquelle certaines clés sont utilisées ou mises à jour. Cela peut être essentiel pour la gestion des clés dans les systèmes où les clés sont régulièrement renouvelées pour des raisons de sécurité.

**Matériaux de Clé Auxiliaires** : Cet identifiant est également lié aux matériels de clés auxiliaires, comme les clés publiques de Paillier, qui sont utilisées pour le chiffrement dans le cadre du protocole. Cela permet de s'assurer que les bonnes clés sont utilisées lors du rafraîchissement.

**Importance :** Le \( epid \) permet de suivre les versions des clés utilisées dans le système, ce qui est crucial pour la sécurité, surtout dans les systèmes qui changent fréquemment de clés pour éviter les attaques.

**Identifiant de Session de Signature (sgid)**

**Description :** L'identifiant de session de signature \( sgid \) est associé spécifiquement à une instance de signature d'un message particulier.

**Composantes Associées :** (Message à Signer) : Chaque \( sgid \) est lié à un message spécifique qui doit être signé. Cela garantit que les signatures peuvent être distinctes même si le même message est signé plusieurs fois.

**Importance :** Le \( sgid \) permet de gérer les signatures pour le même message, offrant une méthode pour distinguer différentes signatures. Cela est essentiel pour éviter les confusions et pour maintenir un audit clair des signatures générées.