



# Monnaies numériques

ESILV 2020/2021



# Ordre du jour



**Wallet management**



**BIP 39**



**BIP 32**

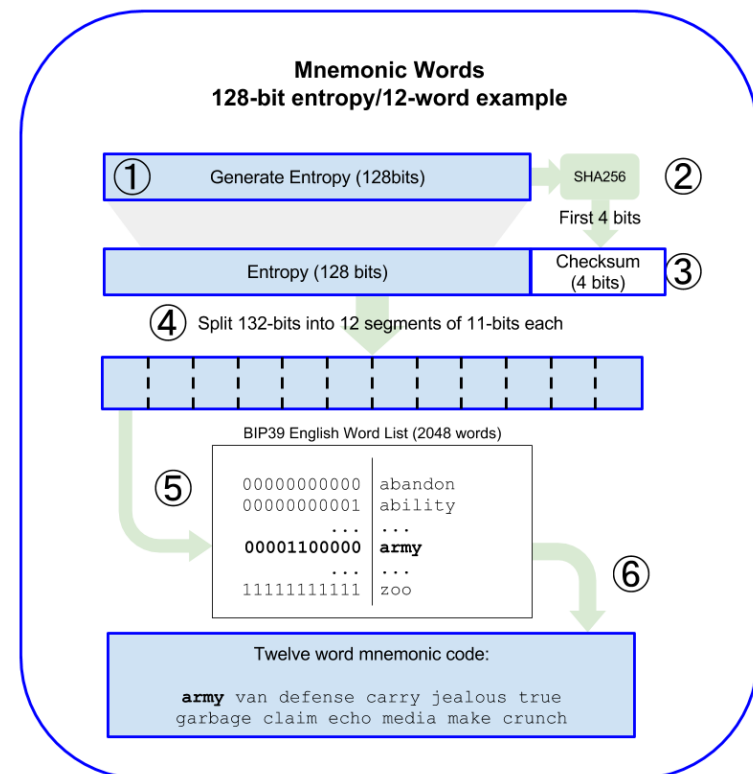


**BIP 39**

# Seeds & Mnemonics

## BIP 39

- La seed d'un wallet est un chiffre aléatoire qui permet de générer les clés du wallet
- Pour faciliter la génération et le stockage de cette seed, BIP39 spécifie un mécanisme permettant de représenter la seed par un ensemble de mots spécifiques
- Différents standards et dictionnaires sont utilisés par différentes cryptos et wallets



# Seeds & Mnemonics

## BIP 39

- <https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki>
- <https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch05.asciidoc>



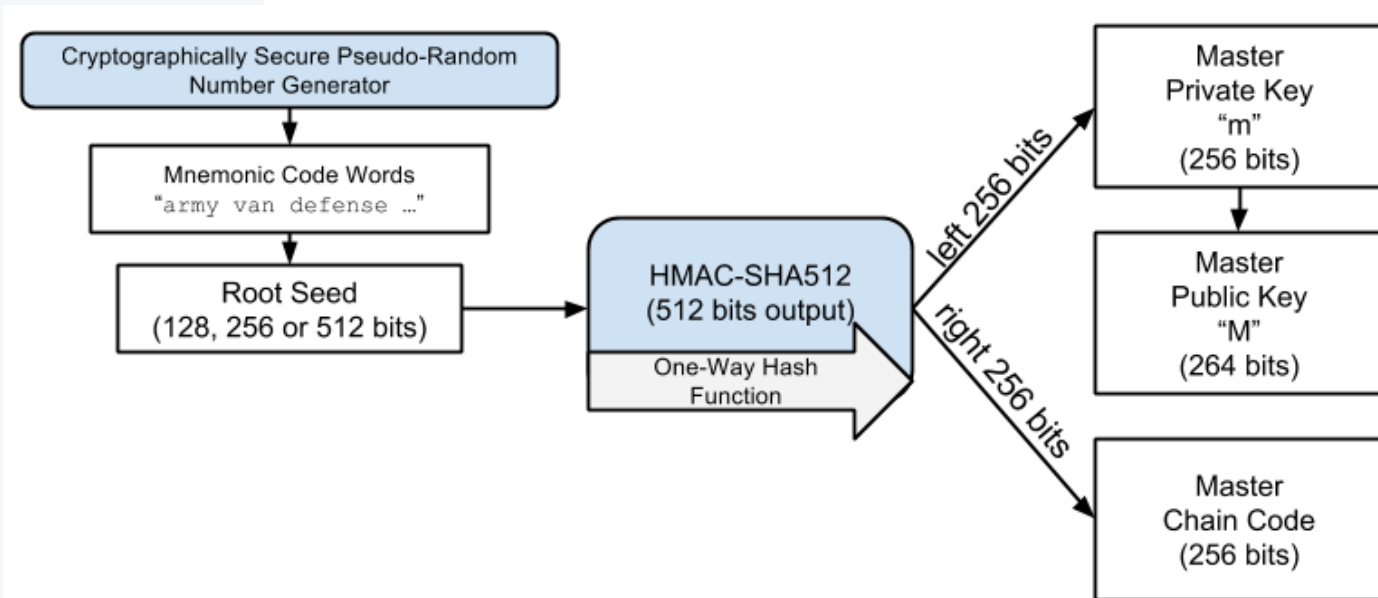
**BIP 32**

# HD Wallets structure

## BIP 43/44

### Récupération d'un wallet à partir de sa seed

- 2 informations sont générées
  - Master private key, pour gérer les fonds avec une première adresse
  - Master chain code, pour introduire de l'entropie et générer les clés suivantes
- Extended public key: Private/public key + chaincode

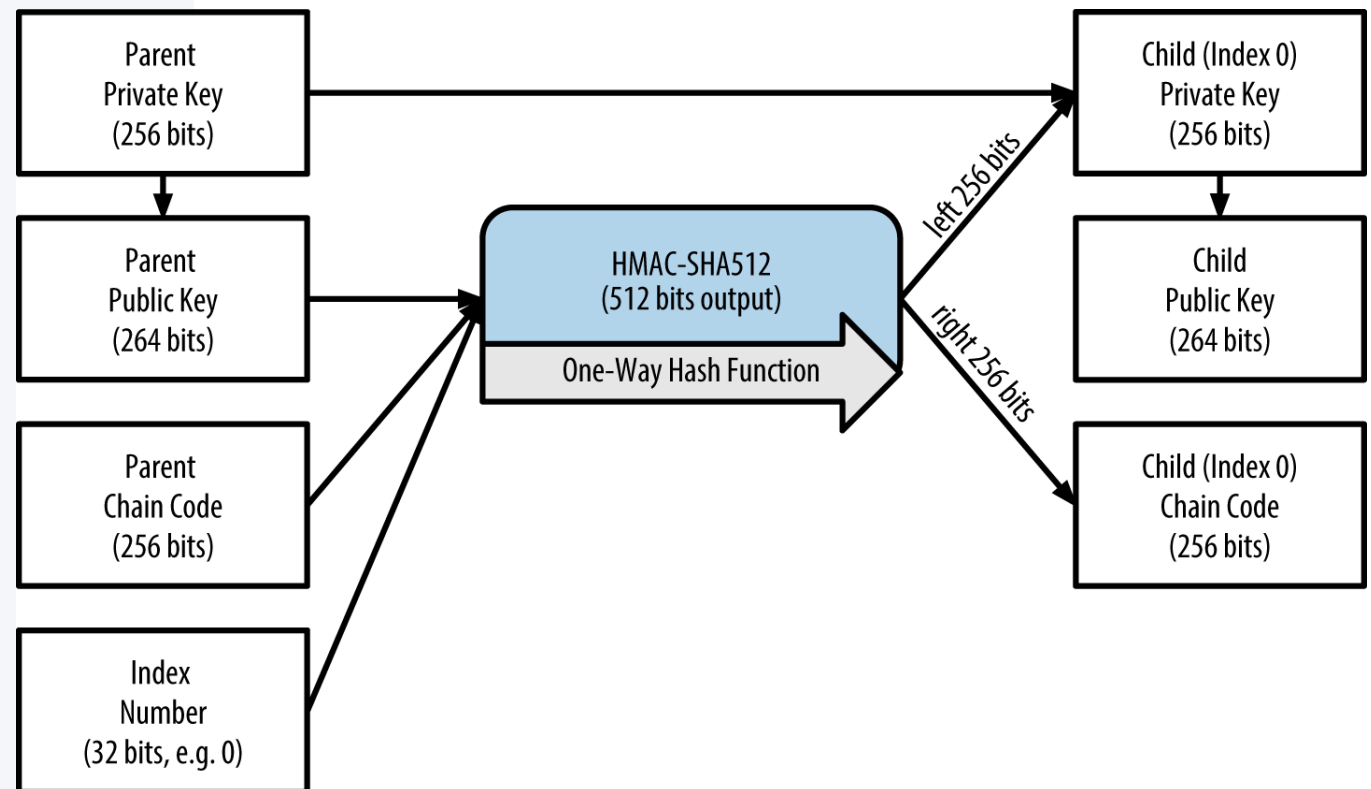


# HD Wallets structure

## BIP 43/44

### Génération de clés enfants

- Chaque génération possède des attributs permettant de générer une génération suivante
- Possibilité de générer l'arborescence à partir d'un Xpub, de façon sécurisée
- Faille potentielle: Compromettre une clé enfant permet de compromettre les parents et l'ensemble du wallet





# Seeds & Mnemonics

## BIP 39

- <https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki>
- <https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch05.asciidoc>



# Tasks list

## BIP 39

- **Créer un repo github et le partager avec le prof**
- **Créer un programme python ou JS interactif en ligne de commande (2pts)**
- **Créer un entier aléatoire pouvant servir de seed à un wallet de façon sécurisée (2 pts)**
- **Représenter cette seed en binaire et le découper en lot de 11 bits (2 pts)**
- **Attribuer à chaque lot un mot selon la liste BIP 39 et afficher la seed en mnémonique (2 pts)**
- **Permettre l'import d'une seed mnémonique (2 pts)**

## BIP 32

- Extraire la master private key et le chain code (2 pts)
- Extraire la master public key (2 pts)
- Générer un clé enfant (2 pts)
- Générer une clé enfant à l'index N (2 pts)
- Générer une clé enfant à l'index N au niveau de dérivation M (2 pts)

# Contraintes

- **Python ou JS**
- **Code exécutable en ligne de commande**
- **Pas de bibliothèques importées  
précompilées, juste des maths**
- **Instructions dans votre readme.md**
- **Rapport en PDF**

## Ressources utiles

- <https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch06.asciidoc>
- Electrum