Blockchain post-quantique

Louis Tremblay Thibault

3 mai 2022

Table des matières

État de l'art et vulnérabilités

- Systèmes de chaîne de blocs utilisent ECDSA pour les signatures numériques
  - Repose sur la difficulté de ECDLP
- Algorithme de Shor rend la résolution de ECDLP facile avec un ordinateur quantique suffisamment puissant
  - Rend les systèmes qui utilisent ECDSA vulnérables !
- Solution : algorithme de signature numérique résistant aux ordinateurs quantiques !

Schéma de signature de Merkle

Suppositions cryptographiques :

- Existence d'un schéma de signature unique (une seule signature par paire de clés) ;
  - Winternitz One Time Signature (W-OTS)
- Existence d'une fonction de hachage résistante aux collisions
  - SHA-256 ; $H : \{0,1\}^* \rightarrow \{0,1\}^{256}$

Winternitz-OTS - génération des clés

W-OTS introduit un paramètre de compromis entre le temps et la taille des clés et des signatures.
On choisit $w \geq 2$ le nombre de bits à signer en même temps et on calcule, avec $n = 256$,

$$t_1 = \left\lceil \frac{n}{w} \right\rceil, \quad t_2 = \left\lceil \frac{\lfloor \lg t_1 \rfloor + 1 + w}{w} \right\rceil, \quad t = t_1 + t_2. \tag{1}$$

La clé privée est choisie au hasard :

$$S = (s_{t-1}, \ldots, s_1, s_0) \in \{0,1\}^{(n,t)}. \tag{2}$$

et la clé publique dérivée est

$$P = (p_{t-1}, \ldots, p_1, p_0) \in \{0,1\}^{(n,t)} \tag{3}$$

où $p_i = H^{2^w - 1}(s_i)$.

Winternitz-OTS - génération de signature

On a un message M et son *digest* $d = H(M)$. On décompose d en t_1 chaînes de bits de longueur w :

$$d = b_{t-1}||b_{t-2}||\ldots||b_{t-t_1} \tag{4}$$

puis on calcule une somme de contrôle sur ces chaînes de bits :

$$c = \sum_{i=t-t_1}^{t-1} (2^w - (b_i)_{10}) \tag{5}$$

et finalement, on complète notre série de t chaînes de bits de longueur w en posant

$$c = b_{t_2-1}||b_0. \tag{6}$$

Winternitz-OTS - génération de signature - suite

Rappel :

$$d = b_{t-1}||b_{t-2}||\ldots||b_{t-t_1} \tag{7}$$

$$\left(\sum_{i=t-t_1}^{t-1}(2^w - (b_i)_{10})\right)_2 = b_{t_2-1}||b_0 \tag{8}$$

Pour obtenir la signature du message M, on applique la fonction de hachage un certain nombre de fois à chaque chaîne de bits de la clé privée :

$$\sigma = (\sigma_{t-1}, \ldots, \sigma_1, \sigma_0) \in \{0,1\}^{(n,t)} \tag{9}$$

où $n_i = (b_i)_{10}$ et $\sigma_i = H^{n_i}(s_i)$.

Winternitz-OTS - vérification de signature

Étant donné une signature $\sigma = (\sigma_{t-1}, \ldots, \sigma_1, \sigma_0) \in \{0,1\}^{(n,t)}$ et un message M, le vérificateur calcule les t chaînes de bits de longueur w comme à dernière section puis vérifie si

$$(H^{2^w-1-n_{t-1}}(\sigma_{t-1}), \ldots, H^{2^w-1-n_1}(\sigma_1), H^{2^w-1-n_0}(\sigma_0)) \quad (10)$$

$$= (p_{t-1}, \ldots, p_1, p_0). \quad (11)$$

Si la signature est valide, on a

$$\sigma_i = H^{n_i}(s_i) \implies H^{2^w-1-n_i}(\sigma_i) = H^{2^w-1-n_i}(H^{n_i}(s_i))$$

$$= H^{2^w-1}(s(i))$$

$$= p_i.$$

Schéma de signature de Merkle (MSS)

La clé privée d'un schéma de signature de Merkle est un arbre de Merkle

- Arbre binaire dont la valeur des nœuds est le hash de la concaténation des deux nœuds enfants.
- La valeur des feuilles de l'arbre est le hash de la clé publique W-OTS associée.

La clé publique est la racine de l'arbre de Merkle.

Schéma de signature de Merkle (MSS) - génération des clés

- Pour un paramètre h choisi, on pourra signer 2^h messages.
- On génère l'arbre de signature en calculant 2^h paires de clés W-OTS et en calculant les valeurs des nœuds de l'arbre de signature.

Schéma de signature de Merkle (MSS) - génération des clés - suite

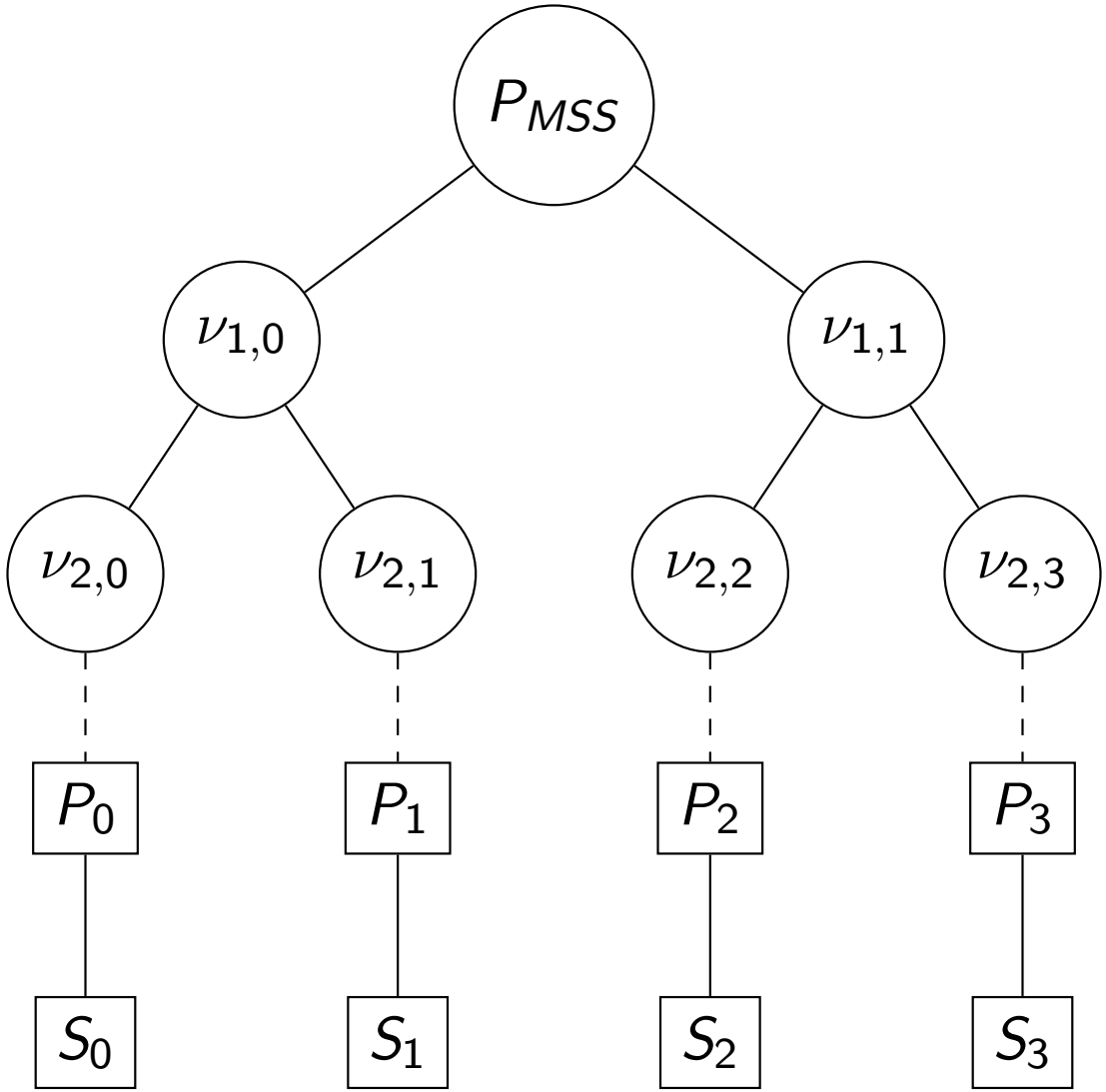

Schéma de signature de Merkle (MSS) - génération de signature

Une signature MSS σ_i est composée d'une signature W-OTS σ_{W-OTS}, de la clé publique P_i associée, de l'index de la feuille i et une preuve de Merkle. On a donc

$$\sigma_i = \{\sigma_{W-OTS}, P_i, i, \{a_1, \ldots, a_h\}\} \tag{12}$$

Pour générer une signature pour une message M, on doit suivre les étapes suivantes :

1. Signer le message M par une signature W-OTS ;
2. Trouver les h nœuds qui formeront la preuve de Merkle.

Schéma de signature de Merkle (MSS) - génération de signature - suite

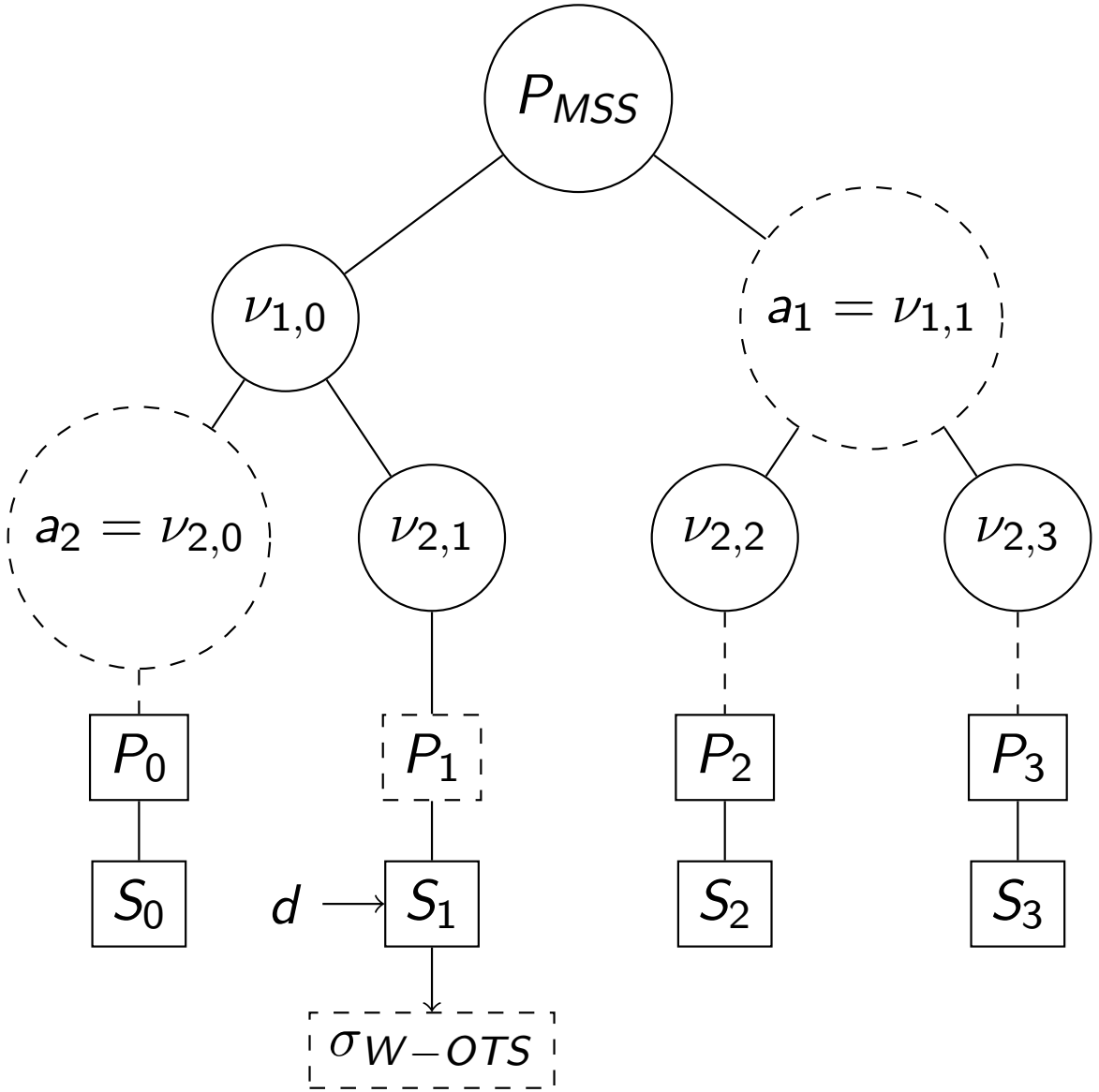

Schéma de signature de Merkle (MSS) - vérification de signature

Étant donné une signature MSS
$\sigma_i = \{\sigma_{W-OTS}, P_i, i, \{a_1, \ldots, a_h\}\}$, un agent vérificateur peut la vérifier (et en dériver la clé publique P_{MSS} du signataire) en suivant les étapes suivantes :

1. Vérifier la signature W-OTS et comparer le résultat avec P_i ;
2. Calculer le chemin de l'arbre entre la i-ième feuille et la racine de l'arbre avec l'ensemble $\{a_1, \ldots, a_h\}$.

MSS - un schéma de signature post-quantique ?

- L'algorithme de Shor brise ECDLP
- MSS n'a pas besoin de ECDLP, mais seulement d'une fonction de hachage résistante aux collisions !
- L'algorithme de Grover trouve des collisions en temps $\mathcal{O}(\sqrt{n})$ plutôt qu'en temps $\mathcal{O}(n)$
- Avec SHA-256, $n = 2^{256}$ et l'algorithme de Grover pourrait trouver une collision en 2^{128} itérations
- Solution facile : utiliser SHA-512

$|coin\rangle$, prototype de blockchain

- Développé entièrement en Go, code original
- Utilise la librairie AMSS, dévelopée dans le cadre du projet
- N'utilise pas de librairie externe
- Mécanisme de consensus Proof-of-Work (comme Bitcoin)
- Système de compte et solde (comme Ethereum)

Analyse des résultats

	ECDSA (secp256r1)	AMSS
Temps de génération de clé	0	$2^h \cdot t \cdot (2^w - 1) + 2^{h+1} - 1$
Taille de la clé privée	256	$2^{h+1} \cdot t \cdot n + n \cdot (2^{h+1} - 1) + h$
Taille de la clé publique	≈ 257	n
Temps de génération de signature	0	$\leq t \cdot (2^w - 1)$
Taille de la signature	512	$2 \cdot (t \cdot n) + h \cdot n + h$
Temps de vérification de signature	0	$\leq t \cdot (2^w - 1) + h$

* Les temps sont en nombre d'évaluations de H et les tailles en bits.

Analyse des résultats - suite

	ECDSA (secp256r1)	AMSS
Temps de génération de clé	$\approx 2,5128 \times 10^{-5}$	$\approx 397,5089$
Temps de génération de signature	$\approx 2,10596 \times 10^{-4}$	$\approx 0,1374$
Temps de vérification de signature	$\approx 1,51299 \times 10^{-4}$	$\approx 0,2363$

* Les temps sont en secondes.

Conclusion

- $|coin\rangle$ est un prototype fonctionnel de système de chaîne de blocs résistant aux attaques menées par des ordinateurs quantiques.
- Il s'agit d'une innovation dans le domaine ; il n'existe aucune solution de ce genre présentement.