Nouvelle authentification du réseau de télécommunication 5G utilisant la cryptographie quantique et post-quantique à clé dynamique

[1]Rakotondramanana R. S. [2]Randriamitantsoa P. A.

Laboratoire de Recherche Télécommunication, d'Automatique, de Signal et d'Images (LR-TASI)

Equipe d'accueil Doctorale de Télécommunication, d'Automatique, de Signal et d'Images (EAD-TASI)

Ecole Doctorale en Sciences et Techniques de l'Ingénierie et de l'Innovation (ED-STII)

Université d'Antananarivo

BP 1500-Antananarivo

[1]radiarisainanasitraka@yahoo.fr, [2]rpauguste@gmail.com

Résumé :

Pour s'authentifier au réseau, le protocole 5G-AKA (5G-Authentification Key Agreement) utilise une authentification mutuelle entre l'opérateur et l'utilisateur. Cependant, la connaissance de la clé maitresse permet de créer un réseau pirate pouvant capturer les trafics des utilisateurs. Le module QPQ-CD (Quantique et Poste Quantique à clé dynamique) consiste à modifier la norme de 3GPP en changeant la clé à chaque authentification. La partie Quantique sera formée par une technique de confusion QHT (Quantum Hilbert Transform) et QAT (Quantum Arnold Transform). La partie Post-Quantique se focalisera sur les méthodes de hachage multiple. Un optimisateur basé sur le changement d'un bit et un sélecteur basé sur la probabilité efficace perfectionnent le choix de la clé suivante. La probabilité efficace de la clé sera en fonction de probabilité d'extrémité, probabilité de proximité, probabilité de bit changé et probabilité de désordre ou entropie. Une des 5 approches de sélecteur et d'optimisateur de clé sera perfectionnée pour raffiner le résultat. En simulant sur Matlab plusieurs essais d'authentification, la probabilité d'extrémité et de proximité seront toutes les deux supérieures ou égales à 50% et peuvent atteindre jusqu'à 100%. La probabilité selon l'entropie est d'ordre de 100%.La probabilité de bit changé stagne sur 50%. La probabilité efficace de sélection en ce moment-là sera d'ordre de 50%.

Mots clés : 5G-AKA, QPQ-CD, QHT, QAT, Matlab

Abstract :

To authenticate to the network, the 5G-AKA (5G-Authentication Key Agreement) protocol uses mutual authentication between operator and users. However, knowledge of the master key can create a hacker network for capturing traffic of users. The QPQ-CD module (Quantum and Post Quantum Cipherkey Dynamic) consists of modifying the 3GPP standard by adding a recurrent change of this key. The quantum part use QHT (Quantum Hilbert Transform) and QAT (Quantum Arnold Transform). The Post-Quantum part will focus on multiple hashing methods. An optimizer based on a one-bit change and an effective probability selector perfect the choice of the next key. The effective probability will be based on probability of extremity, probability of proximity, probability of a bit changed, and probability of disorder or binary entropy and probability of penalties. One of the 5 approaches of selectors and key optimizer will be perfected to refine the result.On MATLAB, the probability of the extremity and proximity will both be greater than or equal to 50% until 100%. The probability in case of binary entropy and disorder will be close to 100%. The probability of

bit change is always at 50%. The effective probability of selection at this time will be greater than 50%.

Keywords: 5G-AKA, QPQ-CD, QHT, QAT, Matlab

1. Introduction

Le régulateur des télécoms a déterminé trois grandes familles d'usages pour la 5G : mMTC (des communications entre une grande quantité d'objets); eMBB (des connexions Internet en très haut débit) ; uRLLC (des communications ultras fiables pour les besoins critiques, avec une latence très faible). De plus, la cryptographie évolue également vers l'Horizon 2020 avec des concepts robustes contre les attaques des ordinateurs quantiques (Post Quantique Crypto), contre les attaques physiques et contre les consommations d'énergies. L'architecture globale de réseau 5G sera représentée par la Figure 1.

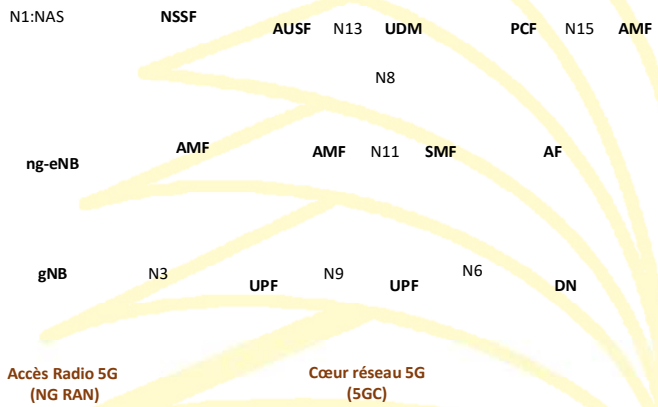

Figure 1 : Architecture globale de réseau 5G

AUSF : Authentication Server Function (AuC)
UDM : Unified Data Management (HLR/HSS)
AMF : Access and Mobility Management Function (MME)
SMF : Session Management Function (MME)
NSSF : Network Slice Selection Function
PCF : Policy Control Function
UPF: User Plane Function (S/PGW)
DN : Data Network (PDN)
AF : Application Function
UE : User Equipment

Les mobiles UE communiquent avec les stations de base par un lien radio 5G. La station de base se nomme Next Generation -eNb (ng-eNb) ou eLTE-eNB. Dans le cœur réseau, on a les blocs suivants :

•Authentication Server Function (AUSF) : traite l'authentification de l'UE

•Core Access and Mobility Management Function (AMF) : Traite la gestion de la mobilité de l'UE

•Policy Control Function (PCF) : traite la gestion de tout type de politique applicable à l'UE (politique de gestion de mobilité, gestion de QoS, gestion de sélection de la technologie d'accès, etc.)

• Session Management Function (SMF) : traite la gestion de session de l'UE

• Unified Data Management (UDM) : Sert d'interface à l'ensemble des fonctions de réseau qui nécessitent accéder aux données de souscription de l'UE.

• User plane Function (UPF) : traite les flux du plan usager sortant et entrants de l'UE

•Application Function (AF) : peut utiliser l'interface PCF pour demander la mise en œuvre de la qualité de service pour un flux IP donné

•Network Slice Selection Function (NSSF) : Permet d'identifier la fonction AMF appropriée pour la prise en charge de la gestion de la mobilité de l'UE.

•Data Network (DN) : concerne les réseaux Data

Figure 2 : Architecture globale de réseau 5G

L'architecture globale de la Figure 1 peut être simplifiée par la Figure 2 en séparant la partie Accès, la partie Cœur et la partie des données. L'algorithme QPQ-CD sera implante à la fois dans l'UE et dans l'AUSPF.

2. 5G-AKA et l'algorithme QPQ-CD

2.1. Procedure 5G-AKA

Les vecteurs d'authentification sont en rouges dans la Figure 3. La carte U-SIM de l'opérateur possède :

- l'identité permanente de la carte SUPI
- la clé maitresse K protégeant l'utilisateur

- la séquence d'authentification SQN pour protéger contre la réutilisation des vecteurs d'authentifications
- la clé publique pk_{HN} pour protéger l'identité de l'utilisateur en OTA (Over The Air) et interopérateur

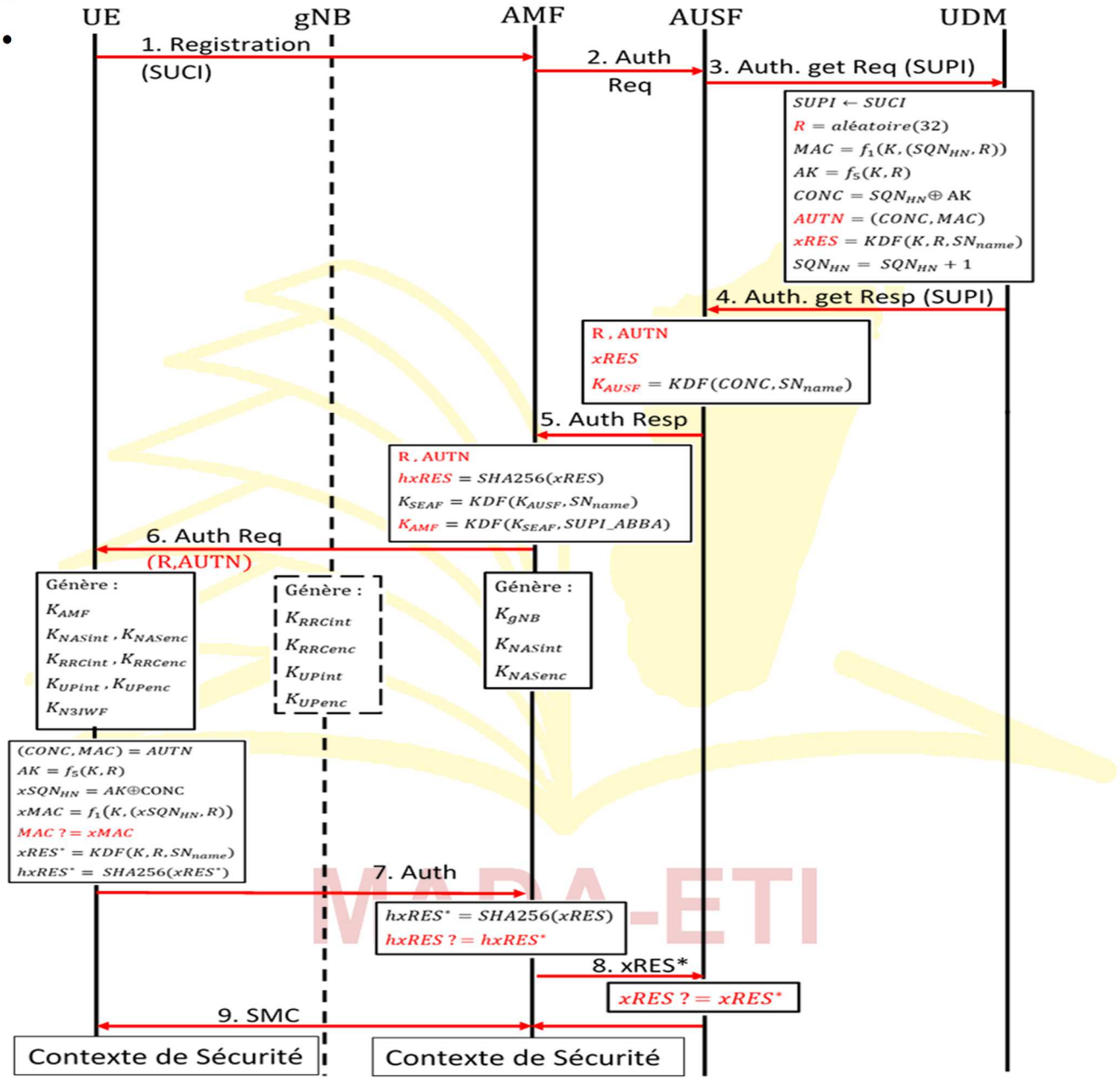

Figure 3 : 5G-AKA

UE : User Equipement et gNB : gigabit NodeB
AMF : Access and Mobility Management Function
AUSPF : Authentication Server Function
UDM : Unified Data Management Function
SUPI : Subscription Permanent Identifier
SUCI : Subscription Permanent Identifier
R : Random Number
MAC : *Message Authentication Code*
K : Master Key
SQN_{HN} : Sequence number of Home Network
AK : Anonimity Key et CONC : CONCEALEMENT

$SUPI, ABBA$: SUPI, Anti Bidding down Between Architectures
K_{gNB} : Key of gNB
K_{NASint} : Key of Non Access Stratum Integrity
K_{NASenc} : Key of Non Access Stratum Security
K_{RRCint} : Key of Radio Ressource Control Integrity
K_{RRCenc} : Key of Radio Ressource Control Security
K_{UPint} : Key Uplink Integrity

La procédure d'authentification AKA-5G (Authentication Key Agreement) suit une norme bien définie par le 3GPP dont la procédure se résume comme suit :

- Pour s'authentifier au réseau, l'UE envoie la signalisation numéro 1 via son SUCI, qui est calculé par un chiffrement public à partir de pk_{HN}, SUPI, R.
- La demande d'authentification vers UDM par la signalisation numéro 3 permet de générer les vecteurs d'authentifications formés par : $RAND, AUTN, xRES$.
- L'AUSPF génère à son tour la clé du serveur et forme les vecteurs d'authentification formés par : $RAND, AUTN, xRES, K_{AUPF}$
- Dans le module AMF après la signalisation numéro 5, les vecteurs d'authentifications sont formés par : $RAND, AUTN, hxRES, K_{AMF}$.
- Pour la partie OTA, le gNodeB envoie ensuite les 2 partie du vecteur d'authentification $RAND, AUTN$ dont $AUTN$ permet à l'UE de vérifie l'authenticité de l'opérateur et le RAND pour donner une réponse à l'opérateur si l'UE est authentique à l'aide de $hxRES$.
- Pour que l'équipement terminal et l'opérateur s'assurent mutuellement de l'interconnexion : l'UE vérifie s'il s'authentifie vraiment au véritable réseau de l'opérateur et non à un réseau pirate à l'aide de la vérification de MAC et parfois de SQN_{HN}. L'UE calcule ensuite $hxRES$ le résultat pour assurer son authenticité.
- Le résultat $hxRES$ sera envoyé à l'AMF et sera vérifié à son tour et $xRES$ sera verifié par l'AUSPF.

- Une fois authentifié, le réseau mobile envoie la signalisation SMC (Session Message Context) pour confirmer l'établissement d'interconnexion. L'algorithme QPQ_CD de la part de l'émetteur utilise le message SMC_ACTIVATION_UE de la part de l'utilisateur et SMC_ACTIVATION_AUSF de la part de l'opérateur dénommé par l'entrée A dans QPQ-CD.

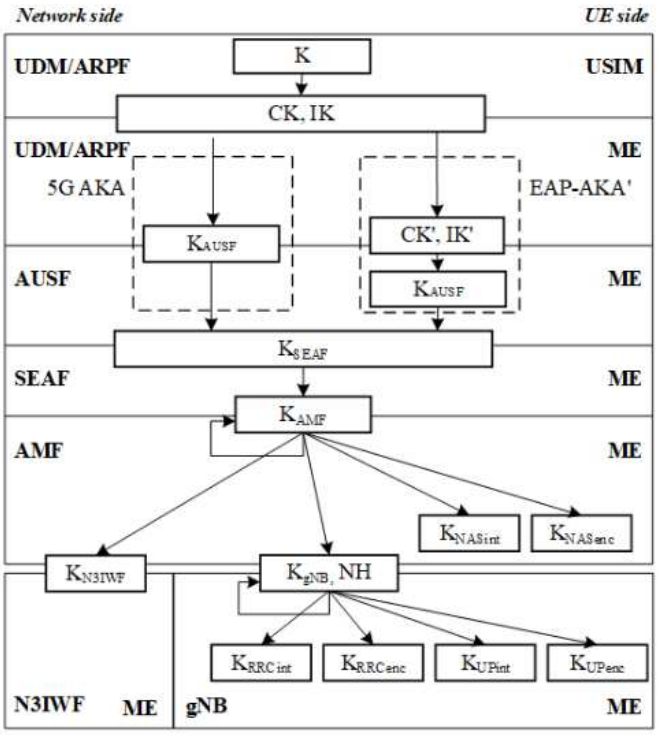

Figure 4 : Hiérarchie des clés 5G

Toutes les clés citées dans la Figure 4. La génération de ces clés sera assurée par la fonction KDF (Key Dérivation Fonction) dont en entrée possède une clé K et certains paramètres de modifications et en sorties la clé dérivée. Toutes les clés dérivent ainsi de la clé maitresse K. EAP-AKA (Extensible Authentication Protocol) permet l'authentification d'un UE-4G dans le réseau 5G.

Les étapes de l'algorithme sont :

- La phase initialisation : le but est d'initialiser K, r, et i un compteur d'activation et générer à partir de la fonction Expansion Matricielle (E.M) une matrice de $16r \times 16r$ de 8bits
- La phase insertion : elle consiste à insérer de manière périodique tout en balayant la ligne de la matrice $16r \times 16r$ une clé obténu par le biais de l'Expansion Linéaire (E.L). L'insertion ne s'exécute qu'à chaque signal d'activation. Comme l'algorithme QPQ-CD utilise 3 matrices $16r \times 16r$, une fonction génératrice de clés notée par G permet de générer 3 parties de sclés d'initialisations pour chaque matrice.

- La phase du crypto quantique : la phase du crypto quantique utilise la méthode de confusion soit par la méthode d'Hilbert soit par la méthode d'Arnold.
- La phase PQ crypto : elle utilise plusieurs échantillons d'algorithme de hachage pour résumer la matrice après confusion afin d'avoir une clé de 256bits.
- Phase sélecrice : elle sélectionne la clé suivante K^+ appropriée.

La sortie de l'algorithme QPQP-CD est une autre clé générée K^+ , pour d' autres applications surtout dans l'authentification. QPQ-CD est donc aussi une famille d'algorithmes KDF. Le schéma simplifié de l'algorithme QPQ-CD est représenté par la Figure 5.

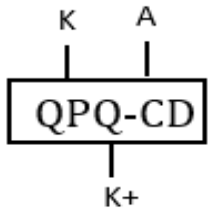

Figure 5 : QPQ-CD

2.2 Expansion, Expansion Linéaire et Expansion Matricielle

L'expansion utilise deux boites selon la Figure 6.

- La boite substitution SBOX
- La tour de modification RCON (Round CONstant)

La module d'expansion modifiée possède en entrée une clé de 16byte, rconj de taille 4.j byte et s_box de 16x16 byte et L_expansion. La sortie produit une matrice linéaire nommée w de taille 16xL_expansion byte.

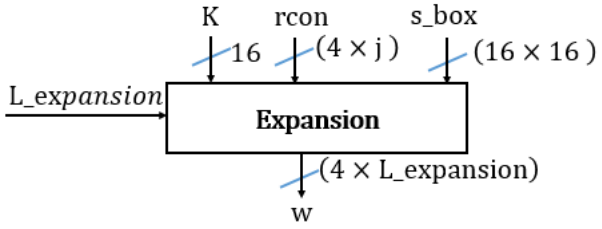

Figure 6 : Diagramme de bloc de l'Expansion

L'algorithme expansion regroupe deux à deux la clé de 128bits pour former un vecteur initial de 16×8bits=16byte. Chaque composant noté par $k_{11}, k_{12}, k_{13}, \quad k_{14}, k_{21}, k_{22}, k_{23}, \quad k_{24}, \; k_{31}, k_{32}, k_{33}, k_{34}, k_{41}, k_{42}, k_{43}, \; k_{44}$ de la matrice représentée par la Figure 7 sera organisé pour former une matrice initiale de 4×4

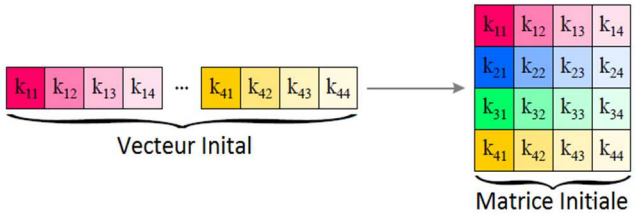

Figure 7 : Initialisation de l'expansion de clé

L'expansion se fait par la Figure 8a qui se répète en utilisant le sous-module d'expansion de la Figure 8b. La première ligne du bloc suivant est obtenue en utilisant le sous-module d'expansion de la Figure 8b et en faisant l'opération xor avec la première ligne du bloc actuel.

La deuxième ligne jusqu'à la quatrième ligne du bloc suivant est obtenue en faisant une opération xor de la ligne précedente du bloc suivant et la ligne du bloc actuel. L'opération est répetée j-fois.

Le bloc assurant le sous module d'expansion possède comme entré rcon_j et clé de 16byte et en sortie $(4 \times L\_expansion)\ byte$. Ce sous module est composé d'une rotation suivie de substitution via la boite s_box et des tours de modification en utilisant l'opération xor à

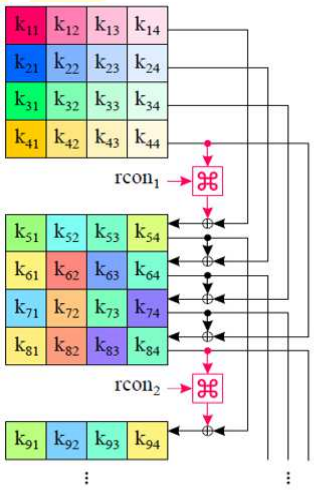

Figure 8a : module d'expansion général

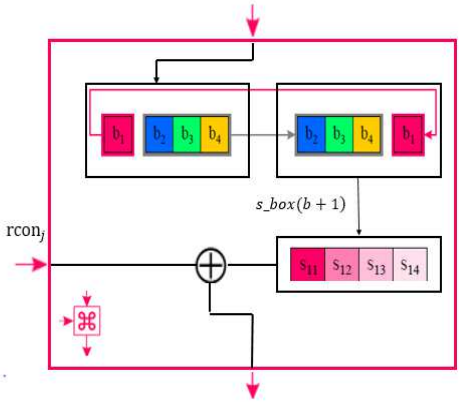

Figure 8b : Sous-module d'expansion

Figure 8 : Schéma bloc d'Expansion de clé complète

L'Expansion Linéaire resp. L'Expansion matricielle possède un paramètre d'entrée K, une clé de 16byte ou de 128bits. Les matrices de modificatoin par tours et de substitution permettant d'étendre la clé en une taille de $4 \times 4r$ resp. $4 \times 64r^2$ en utilisant les modules dans la Figure 9. La module linéarisation permet d'avoir une matrice linéaire de taille 16r resp. $256r^2$. Le module réorganisation transforme la matrice linéaire en une matrice de taille $16r \times 16r$

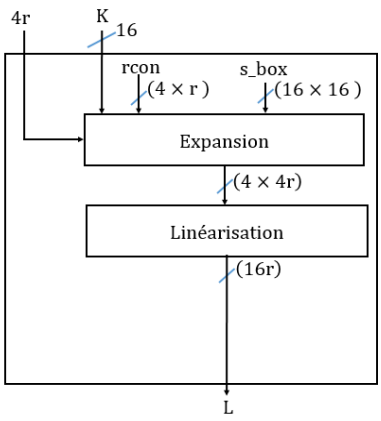

Figure 9a : Expansion linéaire en QPQ-CD

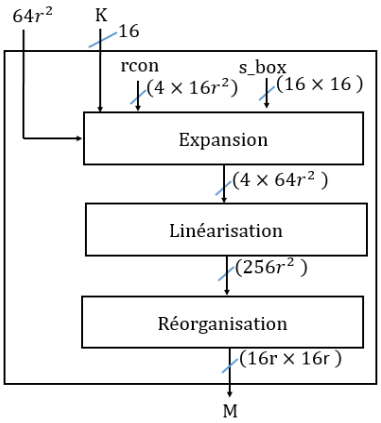

Figure 9b : Expansion Matricielle en QPQ-CD

Figure 9 : Expansion utilisée en QPQ-CD

2.3 Schéma bloc d'un algorithme Génération de clé

Le bloc générateur de clés permet d'avoir trois clés K_1, K_2, K_3 de 128bits (16byte) à partir d'une clé K de 256bits (32byte). Le bloc divise d'abord la clé K en deux clés K1 et K3. K3 est obtenu en faisant xor entre K1 et K2. La Figure 10 représente le bloc de génération de clés

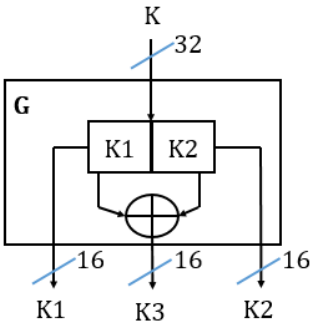

Figure 10 : Schéma bloc de génération de clé

2.4 Algorithme de Q-Crypto (QC)

L'algorithme Q-Crypto [13][14][15] implémente une technique de confusion sur l'image numérique traitée par des processeurs quantiques utilisant QIS (Quantum Image Scrambling). L'entrée de l'algorithme étant une image classique de taille $16r \times 16r$. La matrice sera transformée en modèle FRQI quantique puis traité en QAT (Quantum Arnold Transform) et QHT (Quantum Hilbert Transform). Comme les matrices rouges, vertes, bleus possèdent leur propre composante séparément, les images FRQI correspondantes sont traitées séparément selon les entrées JR, JV, JB. Pour pouvoir traiter par des ordinateurs classiques, le module PQ-Crypto après Q-Crypto, un module de mesure FRQI permet de déterminer le résultat matriciel numérique de qht_R, qat_R, qht_V, qat_V, qht_B, qat_B. Dans l'application générale qht_R, qat_R, qht_V, qat_V, qht_B, qat_B sera simplifiée par q formée par les composants $q_1 \ldots . q_6$

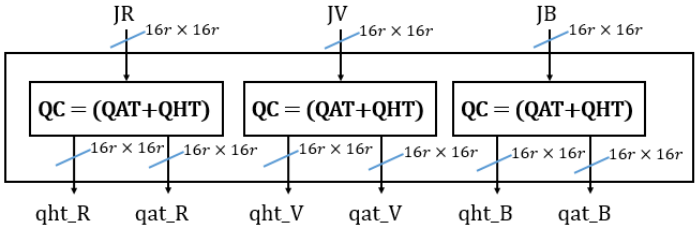

Figure 11 : Module QC

2.5 Algorithme PQ-Crypto

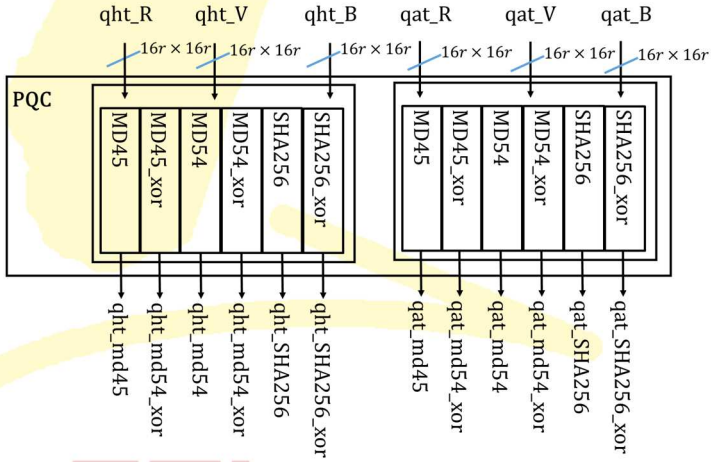

Figure 12 : Module PQC

MD45 : MD4 concaténé avec MD5 sur les matrices en bloc $\_R, \_V, \_B$

MD54 : M5 concaténé avec MD4 sur les matrices en bloc $\_R, \_V, \_B$

SHA256 : SHA 256 sur les matrices en bloc $\_R, \_V, \_B$

MD45 : MD4 concaténé avec MD5 sur les matrices séparées $\_R, \_V, \_B$ suivies de xor entre eux

MD54 : M5 concaténé avec MD4 sur les matrices séparées $\_R, \_V, \_B$ suivies de xor entre eux

SHA256 : SHA 256 sur les matrices en bloc $\_R, \_V, \_B$ suivies de xor entre eux

La matrice en bloc $\_R, \_V, \_B$ est une matrice de trois dimensions de $(16r \times 16r \times 3)byte$ en regroupant les 3 parties qht_R, qht_V, qht_B resp. qat_R, qat_V, qat_B de taille chacune$(16r \times 16r)$. Les 12 sorties $qht\_md45, qht\_md45\_xor, qht\_md54, qht\_md54\_xor$ $qht\_sha256, qht\_sha256\_xor, qat\_md45, qat\_md45\_xor,$ $qat\_md54, qat\_md54\_xor, qat\_sha256, qat\_sha256\_xor$ peuvent être aussi simplifiées par un vecteur p formé par les éléments $p_1 \dots p_{12}$.

2.6. Algorithme de QPQ-CD UE

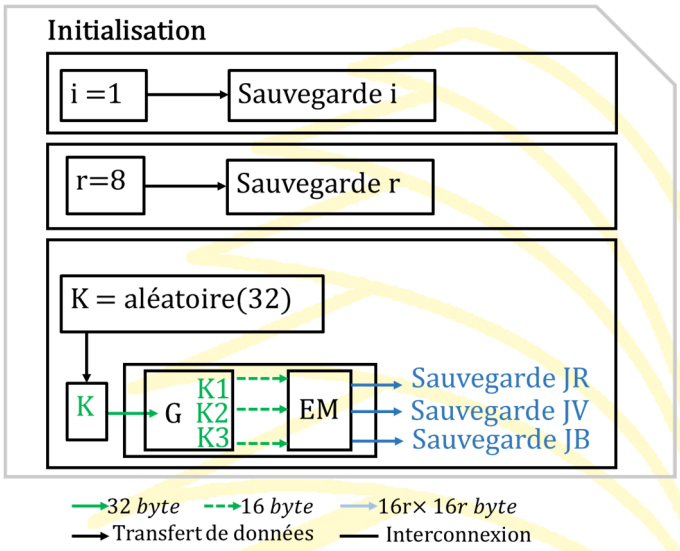

Figure 13 : Initialisation QPQ-CD UE

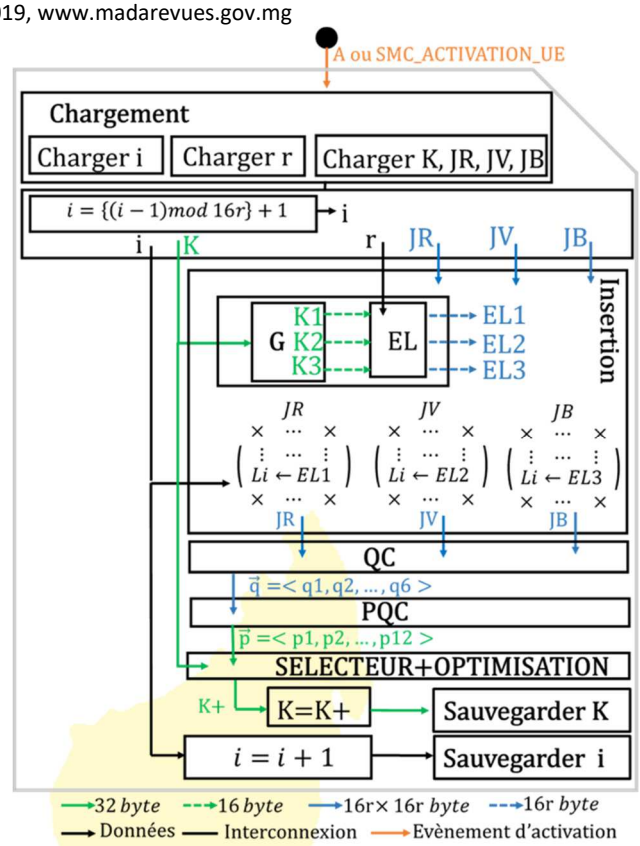

Figure 14 : Activation QPQ-CD UE

La phase d'initialisation selon la Figure 13 est effectuée lors de premier achat de la carte USIM.

- L'algorithme d'initialisation permet d'initialiser d'abord les paramètres i et r ensuite la clé K possédant une valeur de 32byte ou 256bits.
- La clé K sera subdivisée en 3 clés de 16 bytes K1, K2, K3 et l'Expansion Matricielle (E.M.) permet d'avoir les matrices JR, JV, JB de taille $16r \times 16r$

Selon la Figure 14, la phase d'activation est déclenchée à chaque authentification réussie de la part de l'UE. Les paramètres d'initialisations r, i, K et JR, JV, JB seront chargés. La valeur de i est d'abord paramétré pour être dans $1 \dots 16r$.

- Au cours de chaque déclenchement, chaque clé K sera subdivisée en 3 clés K1, K2, K3 et seront transformés par une Expansion Linéaire (E.L) pour avoir EL1, EL2, EL3.
- Ces clés EL1, EL2, EL3 sont insérées successivement dans les matrices JR, JV, JB aux i-ème lignes. La technique de confusion par le module QC permet d'avoir les matrices $q_1 \dots q_6$.
- Des algorithmes PQC de la famille de hachages multiples résument les matrices $q_1 \dots q_6$ pour 12

clés $p_1 \dots p_{12}$ de 32byte ou 256 bit. Le sélecteur et optimisation permet de choisir une seule clé adéquate parmi les 12 clés. La clé sera enregistrée et le compteur d'activation sera incrémenté avant d'être sauvegardé.

2.7 QPQ-CD AUSF

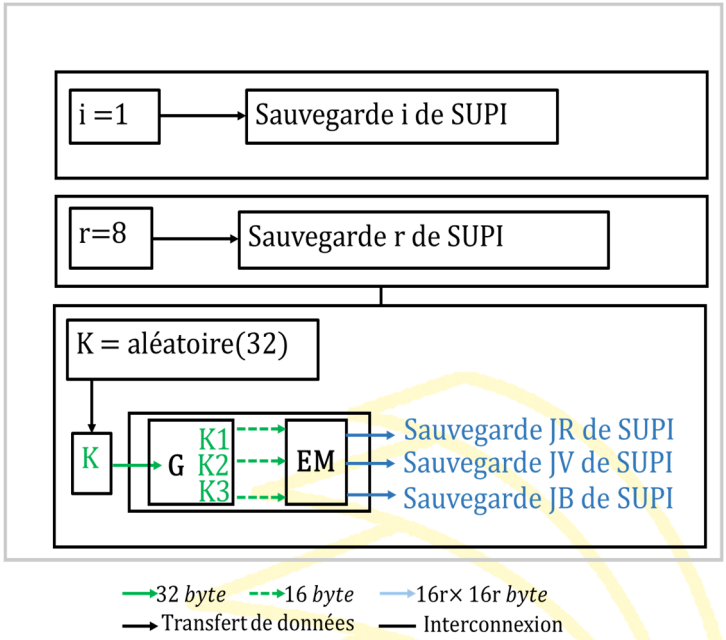

Figure 15 : Initialisation QPQ-CD AUSF

La phase d'initialisation QPQ-CD AUSF est de la même façon que celle de QPQ-CD UE, dans la Figure 13, la différence c'est que l'AUSF gère plusieurs utilisateurs. Les sauvegardes devront être associées à chaque SUPI.

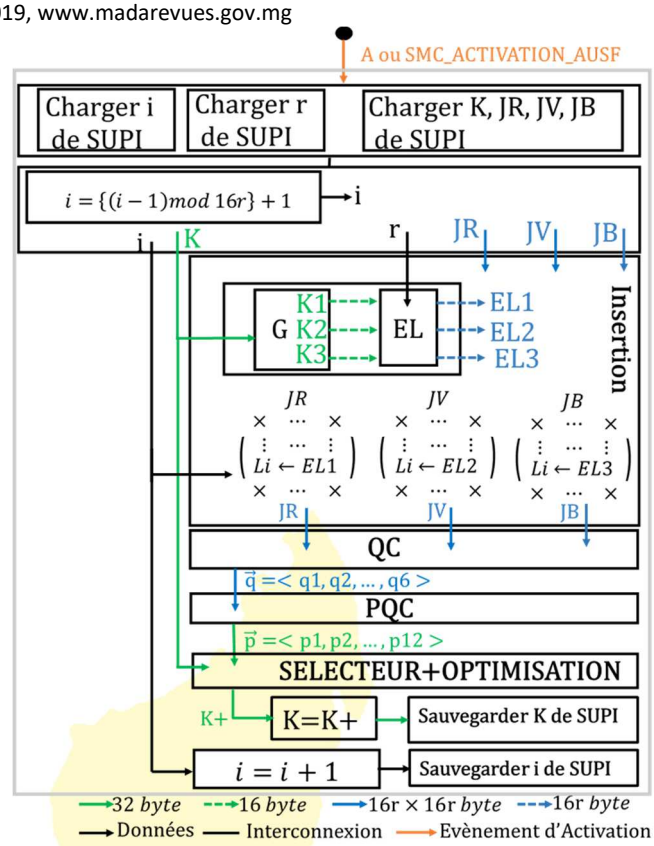

Figure 16 : Activation QPQ-CD AUSF

Pour l'activation selon la Figure 16, la phase d'activation QPQ-CD AUSF est de la même façon que celle de QPQ-CD UE dans la Figure 14, la différence c'est que l'AUSF gère plusieurs utilisateurs. Les sauvegardes et les chargements devront être associés à chaque SUPI.

2.8 Evaluation de QPQ-CD

Pour l'étude de performance de l'algorithme QPQ-CD, le compteur d'activation sera parcouru jusqu'à la fin de la ligne d'insertion. Ainsi, i varie de $1 \dots 16r$.

L'algorithme QPQ-CD sera caractérisé par la phase d'initialisation qui génère r, i, K puis JR, JV et JB. L'insertion des clés générées après l'Expansion Linéaire (E.L) sera fait par une génératrice de clé G et qui sera répétée à chaque boucle jusqu'à $16r$. L'algorithme QC suivi de PQC sera finalisé par le sélecteur d'optimisation pour obtenir la clé suivante $K+$.

L'algorithme de sélection utilise plusieurs critères pour identifier en utilisant la probabilité de ne pas détecter de la clé à partir de la clé précédente en se focalisant sur la manière dont les adversaires pensent, sur d'autres critères pertinents. Vu que l'insertion de la matrice se

fait à chaque ligne de 1 à 16r, l'échantillon d'authentification sera limité à cette valeur 16r.

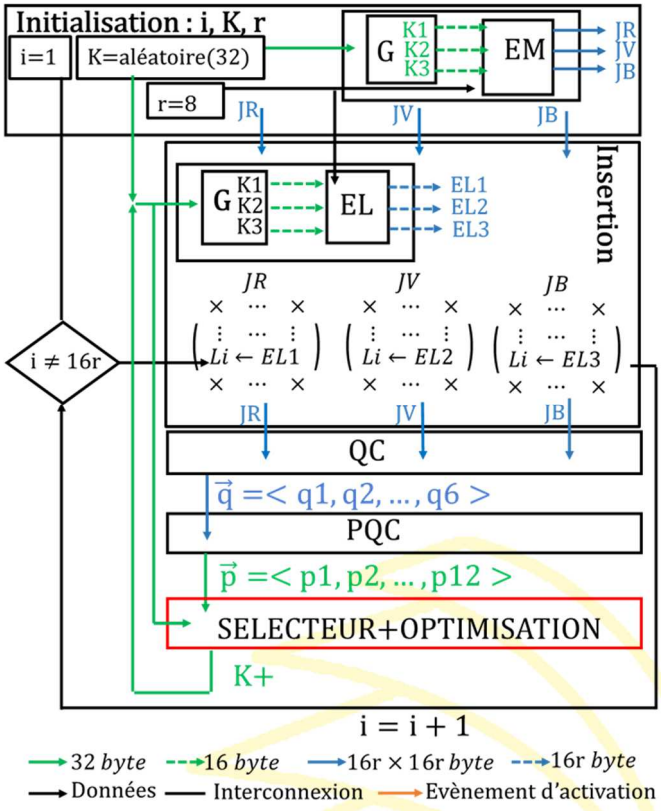

Figure 17 : Evaluation de QPQ-CD

3. Intérprétation sur les approches QPQ-CD

Le sélecteur utilise la probabilité efficace pour sélectionner la meilleure option. La probabilité efficace est tirée de la probabilité d'extrémité, probabilité de proximité, probabilité de bit changé et probabilité de désordre et la probabilité de pénalité. Toutes les courbes étudiées utilisent une interpolation par morceau polynomial d'Hermite connu sous le nom de PCHIP (Piecewise Cubic Hermite Interpolating Polynomial)

- Probabilité d'extrémité :

L'attaque de force brute consistant à parcourir toutes les possibilités de manière aléatoire n'est pas rentable par rapport à la manière ordonnée. Selon la logique ainsi, un adversaire voulant tester toutes les clés possibles en utilisant l'algorithme de la brute force commence toujours par 000…..0 jusqu'à 1111….1 en utilisant l'incrémentation ou en commençant par 111…..11 jusqu'à 0000…..0 en utilisant la décrémentation.

$$\begin{cases} 00000 \dots \dots \dots 000 \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ 11111 \dots \dots \dots .111 \end{cases} \quad \begin{cases} 11111 \dots \dots \dots .111 \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ 00000 \dots \dots \dots 000 \end{cases} \qquad (1)$$

$Incrémentation$ $\qquad$ $Décrémentation$

Plus la clé est proche de 0000….000 ou proche de 111….111 ; plus la probabilité de ne pas détecter la clé est faible.

Si la clé est proche de 0 ; le bit de valeur un des poids forts est difficile à détecter. De ce fait, la probabilité de ne pas détecter la clé si elle est proche de zéro est définie par :

$$p = \frac{\sum_{i=0}^{n-1}[k(i) == 1].2^i}{\sum_{i=0}^{n-1}.2^i} \qquad (2)$$

Si la clé est proche de 1, le bit de valeur zéro des poids forts est difficile à détecter. De ce fait, la probabilité de ne pas détecter la clé si elle est proche du bit un est définie par :

$$q = \frac{\sum_{i=0}^{n-1}[k(i) == 0].2^i}{\sum_{i=0}^{n-1}.2^i} \qquad (3)$$

En utilisant les deux approches, la probabilité pour que la clé soit proche de 0000…000 et de 1111….11 est formée par l'apparition de l'une de deux formules (2) et (3):

$$prob_{extr} :$$

$$= \begin{cases} p = \frac{\sum_{i=0}^{n-1}[k[i] == 1].2^i}{\sum_{i=0}^{n-1}.2^i} \; si \; proche(k, 0000 \dots 000) = 1 \\ q = \frac{\sum_{i=0}^{n-1}[k[i] == 0].2^i}{\sum_{i=0}^{n-1}.2^i} \; si \; proche(k, 1111 \dots 111) = 1 \\ \begin{cases} proche(k, 0000 \dots 000) = (k[n] == 0) \\ proche(k, 1111 \dots 111) = (k[n] == 1) \end{cases} \end{cases}$$

n étant la taille de la clé.

prob_extr étant la probabilité pour que clé k soit proche de l'extrême 0000…000 ou 1111…111

La fonction proche est donnée dans la Formule (5.06)

- Probabilité de proximité

La probabilité de proximité est utilisée sur les deux clés : clé actuelle et clé suivante sont d'autant plus proches l'une sur l'autre. En imaginant deux clés spécifiques à comparer :

$$(k_1, k_2) = (0010{,}0100)$$

La distance entre les deux binaires étant la soustraction entre les deux clés :

$$xor(k_1, k_2) = 0110$$

Pour aller de $k_1 \rightarrow k_2$ sera équivalent à aller de $0000 \rightarrow xor(k_1, k_2)$
Pour aller de $k_2 \rightarrow k_1$ sera équivalent à aller de $1111 \rightarrow xor(k_1, k_2)$

$$prob\_prox = prob\_extr\big(xor(k_1, k_2)\big) \qquad (4)$$

- Probabilité de bit changé

En supposant deux (k_1, k_2) une couple de clé actuelle et clé suivante, l'opérateur xor permet également de vérifier si deux clés ne sont identiques pas identiques.

$$prob_{change} = \begin{cases} \frac{\sum_{i=0}^{n-1} xor(k_1,k_2)[i]}{n} & si \ \frac{\sum_{i=0}^{n-1} xor(k_1,k_2)[i]}{n} \leq 0.5 \\ \left|1 - \frac{\sum_{i=0}^{n-1} xor(k_1,k_2)[i]}{n}\right| & sinon \end{cases} \qquad (5)$$

Entropie : elle est définie par :

$$H = -p(0)log_2(p(0)) - p(1)log_2\big(p(1)\big) \qquad (6)$$

3.1 La probabilité efficace options 1

$$prob_{eff1} = max$$

$$\left\{\frac{4 * prob_{extr} + 3 * prob_{prox} + 2 * prob_{change} + prob_H}{4+3+2+1}\right\}$$

La probabilité efficace sera obtenue en utilisant une évaluation selon la pondération suivant les différents critères en utilisant la moyenne et la combinaison linéaire.

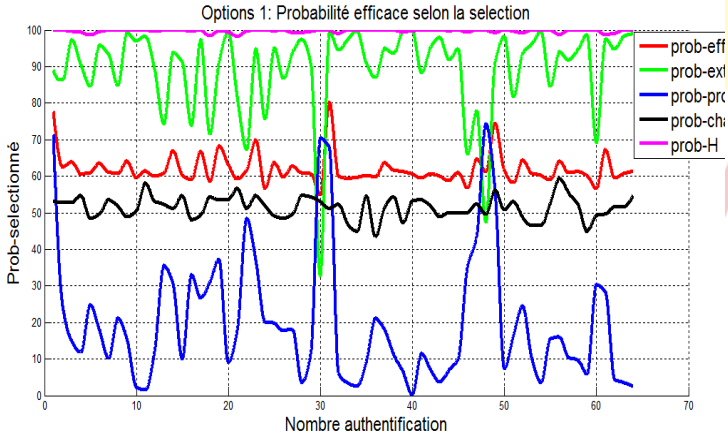

Figure 18 : Probabilité efficace selon option 1

Interprétation :

En tenant compte la priorité de la probabilité de proximité. La probabilité efficace est supérieure à 50%. Cependant, en analysant la probabilité d'extrémité ou la probabilité de proximité, plusieurs cas présentent entre 10% à 30% alors que cette clé est choisie.

Si la probabilité d'extrémité resp. de proximité est très élevée, même si la probabilité de proximité resp. d'extrémité est très faible, le sélecteur choisit encore cette clé au lieu d'autre meilleure option. Dans la Figure 18, l'option 2 permet aussi d'avoir une probabilité de désordre proche de 100% mais probabilité de trouver de bit changé d'environ 50% seulement.

3.2 La probabilité efficace options 2

L'option 2 consiste à prioriser la pondération de probabilité de proximité par rapport à l'extrémité.

$$prob_{eff2} = max$$

$$\left\{\frac{4 * prob_{prox} + 3 * prob_{extr} + 2 * prob_{change} + prob_H}{4+3+2+1}\right\}$$

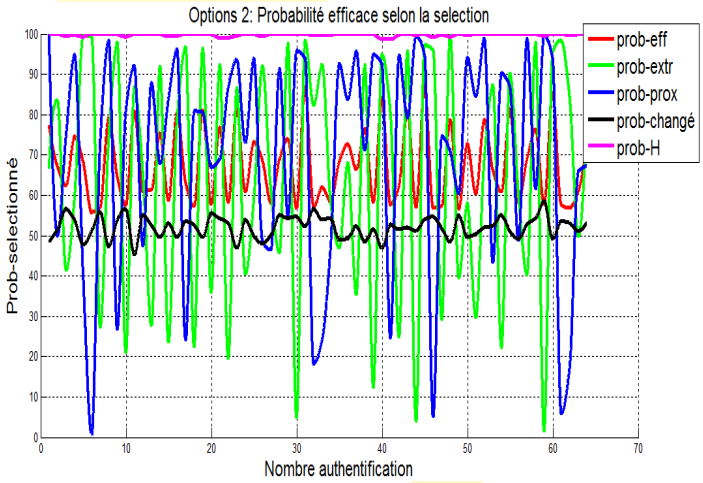

Figure 19 : Probabilité efficace selon option 2

Interprétation :

En tenant compte la priorité de la probabilité de proximité. La probabilité efficace est supérieure à 50%. Cependant, en analysant la probabilité d'extrémité ou la probabilité de proximité, plusieurs cas présentent entre 10% à 30% alors que cette clé est choisie.

Si la probabilité d'extrémité resp. de proximité est très élevée, même si la probabilité de proximité resp. d'extrémité est très faible, le sélecteur choisit encore cette clé au lieu d'autre meilleure option. Dans la Figure 19, l'option 2 permet aussi d'avoir une probabilité de désordre proche de 100% mais probabilité de trouver de bit changé d'environ 50% seulement.

3.3 La probabilité efficace Options 3

Le fait d'étudier les deux paramètres de probabilité d'extrémité et probabilité de proximité séparément n'est

pas une bonne approche. Quand l'une atteint une valeur optimale, l'autre atteint une valeur très faible alors que la probabilité efficace montre une valeur supérieure à 50%. L'option 3 consiste à combiner les deux pour avoir le paramètre $prob_{extr\_prox}$ par :

$$\begin{cases} prob_{eff3} = max\left\{\dfrac{3*\left(prob_{extr\_prox}\right)+2*prob_{change}+prob_H}{3+2+1}\right\} \\ prob_{extr\_prox} = \dfrac{prob_{prox}+prob_{extr}}{2} \end{cases}$$

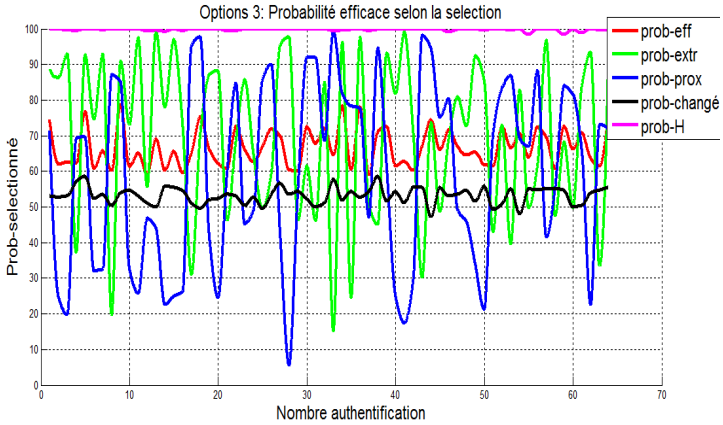

Figure 20 : Probabilité efficace selon option 3

Interprétation :

Selon la Figure 20, la probabilité efficace devient supérieure à 60%, la circonstance montrant que la probabilité d'extrémité resp. Probabilité de proximité proche de 90% engendre que certaine authentification possède de probabilité de proximité ou probabilité d'extrémité inférieure à 20%.

Par rapport à l'option 2, cette valeur minimale n'est plus très proche de zéro. Dans la Figure 20, l'option 3 permet aussi d'avoir une probabilité de désordre proche de 100% mais probabilité de trouver de bit changé d'environ 50% seulement.

3.4 La probabilité efficace Options 4

L'option 4 tient en compte de l'amélioration obtenue en option 3, le but est d'avoir en même temps une probabilité de proximité et probabilité de proximité élevée. Or, même en combinant les deux paramètres, ce cas n'est pas encore résolu. De ce fait, si la moyenne dépasse une valeur de référence 70%, la clé obtenue avec est pénalisée définie par la formule :

$$\begin{cases} prob_{eff4} = max\left\{prob_{eff3} + Penalty(prob_{extr_{prox}})\right\} \\ enalty\left(prob_{extr\_prox}\right) = \begin{cases} 0\ si\ min(x-ref, y-ref) \geq 0 \\ min(x-ref, y-ref)\ ailleur \end{cases} \end{cases}$$

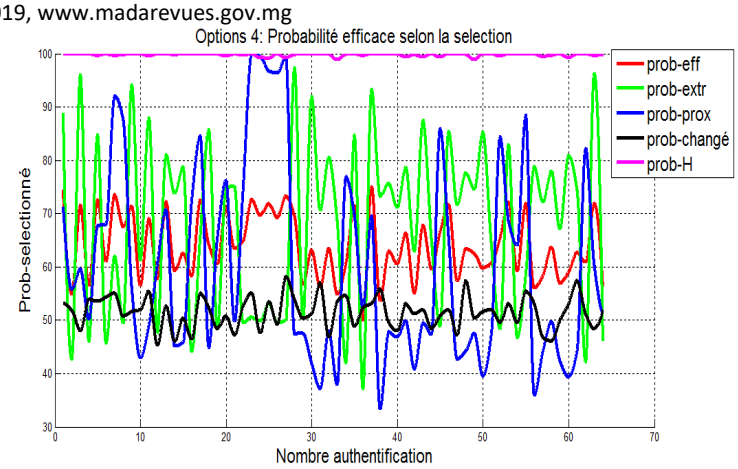

Figure 21 : Probabilité efficace selon option 4

Interprétation :

La probabilité efficace étant supérieure à 50% avec laquelle la probabilité de proximité et la probabilité d'extrémité seront supérieure à 30% toutes les deux avec une probabilité d'entropie d'environ 50% et probabilité de désordre d'environ 100%.

Pour avoir un visuel comparatif de probabilité d'extrémité et de proximité, la Figure 21 a été introduite pour l'analyse de la distance de clé.

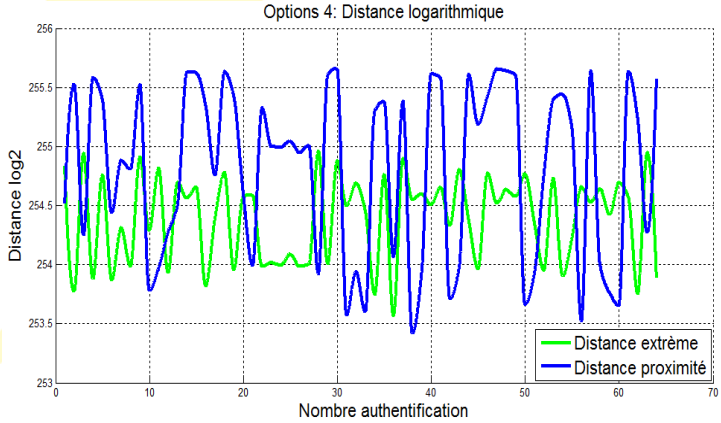

Figure 22 : Distance logarithme selon l'option 4

Interprétation :

Tester toutes les combinaisons d'une clé de 256bits équivaut à visiter l'univers tout entier. Tracer la distance entre les deux clés (précédentes et suivantes) et la distance minimale par rapport à l'extrémité (00…000 ou 11…111) sera représentée par une distance logarithmique en base 2.

La distance logarithmique étant défini par :

$$\begin{cases} d_{prox} = log_2(|(K_+)_{10} - (K)_{10}|) = log_2(|(K_+)\oplus(K)|) \\ d_{extr} = \min\left(log_2(|(K_+)\oplus(00\ldots000)|); log_2(|(K_+)\oplus(11\ldots111)|\right. \end{cases}$$

K_+ Clé suivante après QPQ-CD

K la clé précédente avant le QPQ-CD

d_{prox} distance de proximité entre les deux clés entrantes et sortantes du QPQ-CD

d_{extr} distance des extrémités de clé sortante QPQ-CD

La Figure 22 montre alors que la distance séparant la clé suivante et la clé précédente et la distance séparant la clé aux bordures 00…00 et 11…11 sont distantes d'environ $2^{253.5}$ au minimal et de $2^{255.5}$ au maximal.

Pour l'entropie et la probabilité de bit changé, la Figure 23 permet de mieux représenter l'évolution de ses deux paramètres

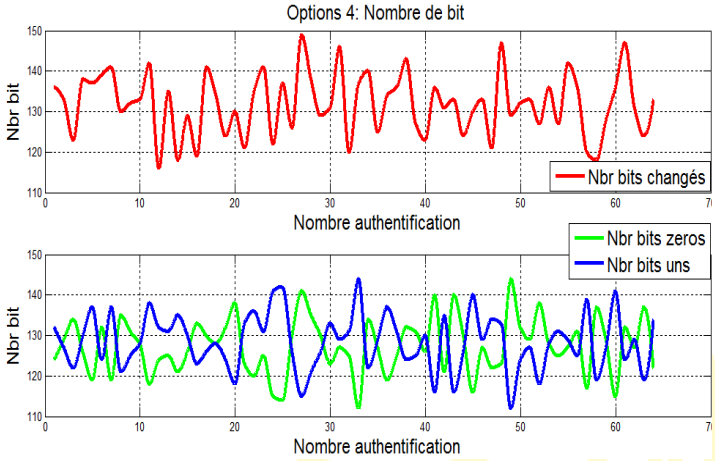

Figure 23 : Nombre de bits selon l'option 4

Interprétation :

Pour mieux visionner la probabilité de désordre de la clé et la probabilité de bit changé, la Figure 23 a été introduite. Le nombre de bits changé varie donc de 115 à 150. Ainsi, le bit de probabilité selon la Figure 21 est de 50% vu que la taille de la clé est de 256bits.

De plus, le nombre de bits de zéros et nombre de bits permet de vérifier la probabilité de désordre Prob-H de la Figure 21. Plus la probabilité d'apparition des zéros et des uns dans la clé est proche de 50% plus la clé est en désordre. De ce fait, le nombre de bits de zéros et des bits un est proche et symétrique au nombre de bits 128 qui est 50% de 256 dans la Figure 23. Le but est d'avoir à la fois la probabilité de désordre proche de 100% et la probabilité de bit changé vraiment égal à 50% pour conduire à l'option 5.

3.5 La probabilité efficace Options 5

L'option 5 consiste à ajouter un optimisateur de sélection avant la sélection lui-même. L'optimisateur consiste à augmenter le nombre de choix de la clé de 256fois en ne changeant qu'un seul bit dans la clé. Ensuite à utiliser un sélecteur de l'option 4. Les 12 clés optimisées sortant du bloc PQC seront ensuite sélectionnées par le même sélecteur de l'option 4. La Figure 24 représente le sélecteur avec optimisation.

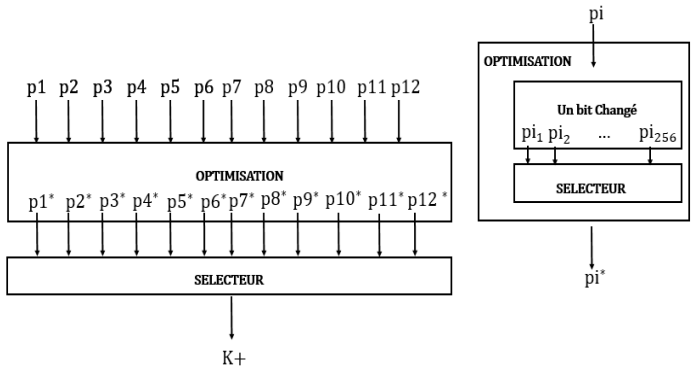

Figure 24 : Schéma bloc d'un selecteur avec optimasion

Chaque clé reçue après le bloc PQC dont les 12 sorties : $qht\_md45, qht\_md45\_xor, qht\_md54, qht\_md54\_xor, qht\_sha256, qht\_sha256\_xor, qat\_md45, qat\_md45\_xor, qat\_md54, qat\_md54\_xor, qat\_sha256, qat\_sha256\_xor$ peut être aussi simplifiée par un vecteur p formé par les éléments$p_1 \dots p_{12}$. L'optimisateur va choisir les meilleurs clés en changeant un bit de modification à chacun de ces clés pour donner$p_1{}^* \dots p_{12}{}^*$. Le sélecteur va choisir une des clés optimisées pour n'avoir qu'une seule clé $K+$

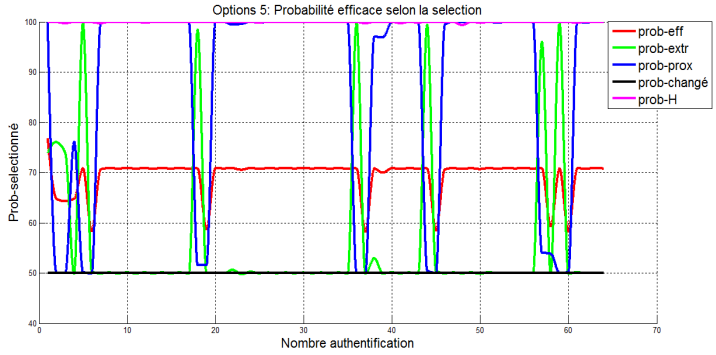

Figure 25 : Probabilité efficace selon l'option 5

Interprétation :

Par rapport à l'option 4, l'optimisateur a pu atteindre jusqu'à environ 80% de probabilité efficace tout en ayant une probabilité de proximité supérieure ou égale à 50% et une probabilité d'extrémité variant entre 68 à 82%. Le plus important encore, au lieu d'avoir une probabilité de bit changé d'environ 50%, pour la Figure 25, elle est vraiment égale à 50% tout en ayant une probabilité de désordre très proche de 100%.

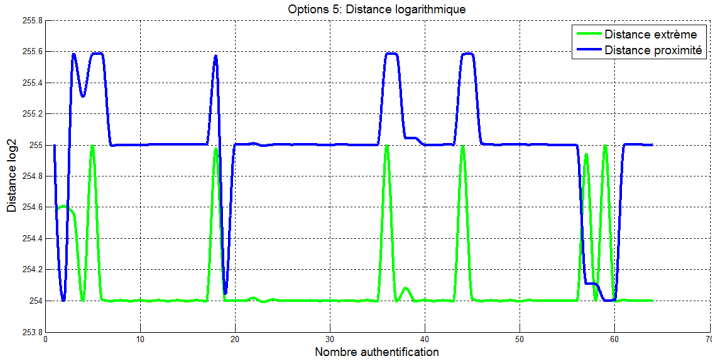

Figure 26 : Distance logarithmique selon l'option 5

Interprétation :

La distance de proximité et la distance d'extrémité sera supérieure à 2^{254} dans la Figure 26. La distance logarithmique est un paramètre très sensible. Dans le cas de l'option, les deux distances minimales sont de $2^{253.5}$. Le fait de changer de distance logarithmique de 0.5, l'optimisateur a pu optimiser d'environ $2^{0.5}$ qui est de 1.414 fois par rapport à $2^{253.5}$ de l'option 4.

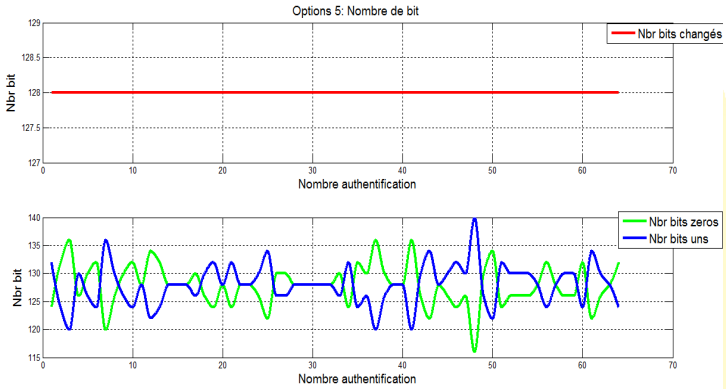

Figure 27 : Nombre de bits selon l'option 5

Interprétation :

Selon la Figure 27, la probabilité de désordre prob-H est d'environs de 100% tandis que la probabilité de bit changé est vraiment de 50%.La Figure 25 montre que le nombre de bits changé égal à 128 qui est une valeur maximale de ce paramètre. Une clé de 50% de bit changé est difficile à cracker que celle de 0% ou de 100%. La courbe représentant le nombre de bits zéros et bits uns sont symétriques par rapport au nombre de bits 128 donc tous les deux sont désordonnés.

3.5 Tableau des approches du sélecteur

Tableau 1 : Approche du selecteur

Options	Pondération				Penalité	Optimiseur
	Prob_prox	Prob_extr	Prob_change	entropie		
1	3	4	2	1	**Non**	**Non**
2	4	3	2	1	**Non**	**Non**
3	3		2	1	**Non**	**Non**
4	3		2	1	**Oui**	**Non**
5	3		2	1	**Oui**	**Oui**

Le Tableau 1 montre que la première approche utilise seulement la pondération par ordre de priorité. Le problème de cette approche sera la probabilité d'extrémité sera très optimale, mais la probabilité de proximité varie de meilleurs état à l'état pires. La deuxième approche est très proche de la première mais la pondération sera inversée entre la probabilité d'extrémité et la probabilité de proximité. Le problème de cette approche est que les deux probabilités d'extrémités et de proximités ne sont pas optimales en même temps. Dans la troisième approche, ces deux probabilités seront étudiées en même temps. Le problème de cette approche est que certaine valeur de probabilité d'extrémité ou de proximité sera très inférieure à 10%.

Pour cela, le quatrième approche utilise le concept de pénalité : si l'une de ces deux probabilités dépasse une certaine valeur, les pénalités permettent de ne pas avoir une valeur très grande, mais assez stable environ de 70% des deux cas. Le problème de cette approche sera que la probabilité de bit changé stagne encore dans le 50%.Le concept d'optimisateur qui change un bit à chaque clé à choisir permet d'augmenter le choix. De ce fait, de désordre sont très proche de 100% et la probabilité de bit changé est égale à son maximal 50%.

4. Conclusion

L'architecture globale du réseau 5G peut être traduite en architecture simplifiée possédant l'accès ; le cœur et les données. Pour s'authentifier au réseau 5G, le protocole AKA utilise d'authentification mutuelle tout en vérifiant si l'opérateur est authentique et si l'utilisateur l'est aussi. La clé maitresse partagée entre l'opérateur et la carte U-SIM sera la base de la sécurité du protocole. Cette clé étant statique. L'algorithme QPQ-CD qui sera formé des algorithmes d'expansion de matrice puis l'algorithme de

confusion utilisant la représentation des images quantiques et se terminera par une un algorithme post-quantique pour avoir une clé de 256bits. Après chaque authentification réussie, la clé maitresse change dynamiquement tout en respectant certains critères : probabilité de bit changé et probabilité selon l'entropie binaire. La dernière approche tenue en compte utilise la méthode de gratification par pondération combinée avec un algorithme de pénalisation et optimisée par l'augmentation de choix en ne changeant qu'un seul bit. La pondération se fait par ordre croissant de la probabilité selon l'entropie binaire suivie de bit changé et la moyenne de la probabilité d'extrémité et de proximité. Ces deux probabilités ne peuvent pas être maximales en même, ainsi le sélecteur pénalise les clés générées ayant ses valeurs supérieures à 70%. Après optimisation, la probabilité de proximité et d'extrémité seront toujours supérieure à 50% tous les deux, la probabilité selon l'entropie binaire sera d'ordre de 100%. La probabilité de bit changé sera à sa valeur maximale qui stagne à 50% tout le long de l'authentification.

5. Bibliographies

[1] Y. Wu, H. Huang, C. Wang, Y. Pan, *« 5G Enabled Internet of Thing »,* CRC Press, 2019

[2] V. C. M. Leung, H. Zhang, X. Hu, Q. Liu, Z. Liu, *« 5G for Future Wireless Networks »,* ICST Institute for Computer Sciences, 2019

[3] V. C. M. Leung, H. Zhang, X. Hu, Q. Liu, Z. Liu, *« 5G for Future Wireless Networks »,* ICST Institute for Computer Sciences, 2019

[4] W. Lei, Anthony C.K. Soong, L. Jianghua ,W. Yong , B. Classon, W. Xiao, D. Mazzarese, Z.Yang, T. Saboorian, *«5G System Design An End to End Perspective»*, Springer, 2020

[5] H. Fattah, *« 5G LTE Narrowband Internet of Things »,* CRC Press, 2019

[6] S. M. A. Kazmi, L. U. Khan, N. H. Tran, C. S. Hong, *« Network Slicing for 5G and Beyond Networks »,* Springer, 2019

[7] T. Q. Duong, X. Zhou, H. V. Poor, *« Ultra-dense Networks for 5G and beyond»*, the Institution of Engineering and Technology, John Wiley, 2019

[8] R. Khan, P. Kumar, D. N. K. Jayakody, M. Liyanage, *« A Survey on Security and Privacy of 5G Technologies: Potential Solutions, Recent Advancements and Future Directions»,* Journal of IEEE, Juil. 2019

[9] R. Khan, P. Kumar, D. N. K. Jayakody, M. Liyanage, *« Novel 5G Authentication Protocol to Improve the Resistance against Active Attacks and Malicious Serving Networks»,* Journal of IEEE, Sept. 2019

[10] L. Song, Z. Xu, Z. Tian, J. Chen, R. Zhi, *« Research on 4G And 5G Authentication Signaling»,* International Journal Of Physics, 2019

[11] R. Khan, P. Kumar, D. N. K. Jayakody, M. Liyanage, *« A Survey on Security and Privacy of 5G Technologies: Potential Solutions, Recent Advancements and Future Directions»,* Journal of IEEE, Juil. 2019

[12] R. Borgaonkar, L. Hirschi,, S. Park, and A. Shaik, *« New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols»,* Journal of Sciendo, 2019

[13] H. Liu, B. Zhao, L. Huang, *«Quantum Image Encryption Scheme Using Arnold Transform and S-box Scrambling»*, Journal of MDPI, Mar. 2019

[14] S. Heidari, M. Houshmand, N. T. Mashadi, *«A dual quantum image scrambling method»,* Quantum Information Processing, Jan. 2019

[15] M. Heigly, M. Schrammy, D. Fiala, *«A Lightweight Quantum-Safe Security Concept for Wireless Sensor Network Communication»,* Journal of IEEE 2019

[16] M S. Shoba, *«A Survey on Post Quantum Digital Signature Schemes for Blockchain»,* International Journal of Computer Science and Mobile Computing, June 2019