Etat de l'art sur les mécanismes de contrôle d'accès dans les réseaux de capteurs sans fil

Youssou FAYE
Département Maths-Info
Univ. C. A. Diop de Dakar
SENEGAL
fayoussouf@yahoo.fr

Ibrahima NIANG
Département Maths-Info
Univ. C. A. Diop de Dakar
SENEGAL
iniang@ucad.sn

Thomas Noel
Département Maths-Info
Univ. strasbourg
FRANCE
noel@unistra.fr

RÉSUMÉ. Le contrôle d'accès devient une nécessité pour empêcher les entités non autorisées à joindre le réseau de capteurs sans fil. En effet, d'une part, le réseau de capteurs doit être en mesure de donner et de retirer l'autorisation d'accès aux utilisateurs. D'autre part, le réseau de capteurs doit être organisé afin d'assurer l'authentification de requêtes. Donc, un adversaire ne doit pas avoir la possibilité d'envoyer des requêtes sur les données collectées par les capteurs. Pour cela, la solution de sécurité doit restreindre l'accès seulement aux capteurs et utilisateurs éligibles, tandis que les requêtes venant des entités non autorisées ne sont ni traitées ni transmis par les capteurs dans le réseau. Dans cet article, nous proposons une étude des solutions de contrôle d'accès dans les réseaux de capteurs sans fil. Ainsi, nous examinons différents algorithmes pour mettre en évidence leurs objectifs, dispositifs, complexité, limites, etc. Nous comparons également ces algorithmes de contrôle d'accès basés sur la densité et la nature du réseau, quelle soit plate ou hiérarchique.

ABSTRACT. To prevent unauthorized entities from joining the wireless sensor network, access control is required. On one hand, wireless sensor networks must be able to authorize and grant users the right to access it. On the other hand, WSN must be organizing for authentication of queries. Thus, an adversary cannot make arbitrary queries on data collected by sensor nodes. On those, the authentication scheme must only restricts the network access to eligible users and sensor nodes, while query from outsiders will not be answered or forwarded by sensor nodes in the network. In this paper, we propose a study of access control solutions in WSNs. Thus, we examine different algorithms so as to find out their objectives, provision, communication complexity, limits, etc. Using the node density parameter, we also provide a comparison of these existed access control algorithms based on the network topology which is flat or hierarchical.

MOTS-CLÉS : Réseaux de capteurs sans fil, Contrôle d'Accès, Authentification.

KEYWORDS: Wireless Sensor Network, Access Control, Authentication.

1. Introduction

Les réseaux de capteurs sans fil (RCSF) sont caractérisés par un déploiement très dense et à grande échelle de nœuds capteurs limités en termes de ressources : capacité à communiquer par diffusion radio à portée réduite, faibles mémoire et capacité de calcul. Leurs domaines d'application restent très variés: la santé, l'agriculture, la surveillance dans les milieux hostiles, etc. Cependant, beaucoup de facteurs rendent ce type de réseau très vulnérable. Parmi ceux-ci, on peut noter : les contraintes d'énergie, les environnements hostiles dans lesquels ils sont déployés, l'absence de sécurité physique et la nature vulnérable des communications radios. Ainsi, l'authentification et le contrôle d'accès, constituent des services de sécurité indispensables aux réseaux de capteurs.

En effet, le contrôle d'accès définit les politiques selon lesquelles les entités (capteurs ou utilisateurs) doivent accéder au RCSF. Dans ce contexte des RCSF, il est essentiel de limiter l'accès du réseau seulement aux entités éligibles, tandis que les messages provenant des nœuds externes (non autorisés) ne devraient pas être transmis dans le réseau. Les données critiques, doivent être protégées contre toute utilisation frauduleuse et accessible à temps réel non seulement depuis la station de base ou les passerelles du réseau, mais parfois aussi par n'importe où dans le réseau à travers les nœuds capteurs en mode ad hoc.

Dans cet article, nous abordons la problématique de sécurité des RCSF basée sur une topologie plate ou hiérarchique permettant la gestion de l'accès. Après une présentation des différents mécanismes de contrôle d'accès, nous proposons une étude comparative des protocoles en se basant sur des critères comme la densité et la nature du réseau. Notons que cette classification s'appuie sur trois catégories: les mécanismes d'admission d'un nouveau nœud, d'authentification d'utilisateurs, et ceux d'authentification de requêtes.

Le reste de ce papier est organisé comme suit : la section 2 présente les topologies des RCSF. La section 3 décrit les issues et défis du contrôle d'accès. Les modèles de contrôle d'accès sont décrits dans la section 4, et les mécanismes dans la section 5. Ensuite, une étude comparative de ces mécanismes est réalisée dans la section 6. Nous terminons par la conclusion dans la section 7.

2. Topologies des RCSF

On distingue généralement deux topologies dans les réseaux de capteurs : la topologie plate ou horizontale et la topologie hiérarchique.

- *Topologie plate* : tous les nœuds capteurs sont homogènes et identiques sauf le puits.

Les nœuds capteurs sont au même niveau et peuvent se communiquer entre eux. Cette topologie peut être centralisée si toutes les données capturées par les nœuds capteurs sont envoyées vers un nœud central qui les traite avant de les renvoyer vers le puits. Elle est utilisée dans les réseaux de petite densité. Dans la topologie plate distribuée, on a plusieurs nœuds de traitement des données qui peuvent se communiquer entre eux.

- *Topologie hiérarchique* : elle introduit deux types de nœuds : des nœuds simples, et des nœuds plus puissants appelés Cluster-Head (CH) assurant plus de fonctions. Ainsi le réseau est divisé en plusieurs clusters, un CH est élu dans chaque cluster, et va gérer les communications inter et intra cluster. Les communications intra cluster peuvent être multi-sauts, et les données reçues d'un niveau sont traitées par les CHs de ce niveau avant d'être transmises vers le niveau supérieur.

3. Les issues et défis du contrôle d'accès

3.1. Les issues

On peut diviser le contrôle d'accès en deux sous services : *authentification* et l'*autorisation*.

L'*authentification* qui consiste à établir une relation entre une entité et son identité qui est une propriété individuelle et ne peut être forgée ni copiée. On distingue deux types d'authentification : authentification des utilisateurs et celle de requêtes. Dans le premier cas, un utilisateur envoie son nom à un nœud capteur et lui prouve son identité et le capteur vérifie si elle est valide ou non. L'authentification de requêtes, permet de vérifier si une requête provient de la station de base (SB), d'un nœud capteur ou d'un utilisateur légitime. Un réseau de capteurs produit l'authentification de requêtes s'il satisfait les propriétés suivantes (peut être avec une certaine probabilité) :

Sécurité : si dans un réseau de capteurs, un nœud capteur accepte une requête comme légitime, celle-ci provient du réseau ou d'un utilisateur autorisé.

Vivacité : toute requête légitime, sera reçue par tous les nœuds capteurs capables de la traiter afin de donner une réponse à l'entité légitime l'ayant postée.

L'*autorisation* : elle consiste à établir une relation entre un utilisateur et un ensemble de privilèges. Dans une autorisation, un utilisateur envoie son nom avec sa requête, à un capteur, qui vérifie si l'opération demandée est autorisée ou non.

Dans un service de contrôle d'accès, l'authentification et l'autorisation sont combinées dans une seule opération.

3.2. Les défis

A cause des contraintes des nœuds capteurs en termes de ressources, les mécanismes de contrôle d'accès doivent minimiser la consommation de ces ressources.

- Un nombre faible de clés avec des tailles demandant moins d'espace mémoire.

-Des algorithmes de sécurité avec moins de calculs afin de préserver l'énergie du RCSF.

- Un nombre minimum de communications pour établir l'algorithme de contrôle d'accès, car la transmission est l'opération la plus couteuse en termes d'énergie.

- La cryptographie asymétrique est plus flexible pour la gestion de clés et d'un grand nombre d'utilisateurs à authentifier. Cependant, elle est moins pratique dans les RCSF car elle demande plus de calculs.

- Le problème de nœud compromis (i.e. prendre le contrôle d'un nœud), fait que les solutions de contrôle d'accès doivent être liées à plusieurs nœuds capteurs.

- L'authentification de message de bout en bout, utilisant une clé symétrique, demande des opérations de déchiffrement/chiffrement par les capteurs intermédiaires, pour authentifier le message. Ce qui nécessite beaucoup de calculs..

Ainsi, le contrôle d'accès devient un vrai défi dans les réseaux de capteurs sans fil.

4. Modèles de contrôle d'accès

Le modèle de contrôle d'accès est fortement lié aux types de communications et services fournis par le RCSF. Nous distinguons deux modèles représentés dans la *figure 1*. Comme présenté sur la figure1(a), on a un réseau de capteurs qui offre des services aux utilisateurs où la station de base sert de point d'accès pour l'administration et la gestion du réseau. Par contre les capteurs servent de points d'accès aux utilisateurs (ordinateur portable, PDA.). Seuls les utilisateurs souscrits aux services du réseau peuvent accéder aux données et le modèle de livraison des données se fait à la demande des utilisateurs.

La figure1 (b) montre un réseau de capteurs sans utilisateur où les données sont envoyées vers le puits. Le modèle d'acquisition et de livraison de données dépend de l'application utilisée et peut être continu, évènementiel, à base de requête, ou hybride.

De ces deux modèles, on a deux niveaux (*interne* et *externe*) de contrôle d'accès.

Le *contrôle d'accès interne* s'intéresse aux communications entre capteurs et entre capteurs et station de base. Il implique les deux modèles (a) et (b). Le *contrôle d'accès externe* sécurise les communications entre le réseau de capteurs (capteurs et station de

base) et les utilisateurs externes. Ces derniers peuvent, en fonction des services souscrits, envoyer des requêtes aux capteurs voisins. Seule la figure1(a) est concernée.

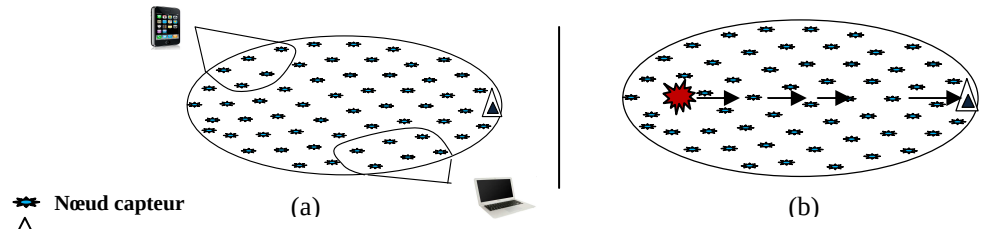

Figure1 : architectures de contrôle d'accès des RCSF

5. Les mécanismes de contrôle d'accès

Dans cette section, nous classons les solutions en trois catégories : les mécanismes de contrôle d'admission, d'authentification de requêtes, et d'authentification d'utilisateurs.

5.1. Les protocoles d'ajout d'un nouveau nœud

A cause de leurs contraintes en ressources et de l'environnement souvent hostile, les capteurs peuvent être attaqués physiquement ou épuiser leur énergie. Par conséquent, l'ajout d'un nouveau nœud capteur devient une nécessité. Beaucoup de solutions sont basées sur la cryptographie asymétrique utilisant les courbes elliptiques. Une première classe des solutions est statique, nous nous intéressons aux solutions dynamiques (utilisées dans les RCSF), c'est-à-dire, l'information existante dans les nœuds capteurs n'est pas mise à jour après l'ajout d'un nouveau nœud capteur.

Nouveau Protocole de Contrôle d'Accès

Cette méthode de H. Huang [1] utilise en plus de la cryptographie asymétrique**,** les chaines de hachages. Elle effectue une authentification et un mécanisme simple de génération de clés pour des nœuds capteurs. Basée sur la méthode de Zhou et al. [2]. chaque nœud exécute des fonctions de hachage et d'opérations OU-exclusif pour accomplir une authentification mutuelle et établir une clé partagée pour sécuriser les communications. Elle est composée de trois phases: une phase d'initialisation, une phase d'authentification et d'établissement de clés, et une phase d'addition de nouveau nœud.

Phase d'initialisation : dans cette phase, la station de base(SB) choisit r clés secrètes $\{k_1, k_2, ..., k_r\}$ pour r nœuds $\{N_1, N_2, ..., N_r\}$ voisins du nœud à ajouter. Puis charge chaque clé K_i dans le nœud N_i correspondant avec la fonction de hachage à sens unique h().

La SB calcul $h^z(k_i) = h(h^{z-1}(k_i))$ et diffuse $h^z(k_i)$, avec z un grand nombre constant, et $h^z(k)$ désigne l'application de la fonction de hachage h() *z* fois sur k.
Phase d'authentification et d'établissement de clés : cette phase s'effectue comme suit pour deux voisins N_i et N_j avec respectivement les chaines $h^{z-u}(k_i)$ et $h^{z-v}(k_j)$:

1) N_i génère un nombre aléatoire t_i, calcule $A_i = t_iP = (Ax_i,Ay_i)$ sur la courbe elliptique et $s_i = h(Ax_i|| h^{z-u-1}(k_i))$ puis diffuse $\{A_i, s_i,N_i\}$. N_j génère un nombre aléatoire t_j , calcule

$A_j = t_jP = (Ax_j,Ay_j)$ sur la courbe elliptique et s_j and $s_j = h(Ax_j|| h^{z-v-1}(k_j))$ puis diffuse $\{A_i, s_i,N_i\}$.

2) N_i calcule $K_{ij} = t_iA_j = (Kx_{ij},Ky_{ij})$ et $z_i = h(Kx_{ij}|| h^{z-u-1}(k_i))$ et diffuse $\{z_i, h^{z-u-1}(k_i)\}$. N_j verifie $h(h^{z-u-1}(k_i)) = h^{z-u}(k_i)$. Si c'est valide, N_j calcule $K_{ij} = t_jA_i = (Kx_{ij},Ky_{ij})$ puis vérifie si $h(Ax_i|| h^{z-u-1}(k_i)) = s_i$ et $h(Kx_{ij}||h^{z-u-1}(k_i)) = z_i$. Si c'est valide, N_i *est* authentifié par N_j.

3) N_j calcule $z_j = h(Kx_{ij}||h^{z-v-1}(k_j))$ diffuse $\{z_j, h^{z-v-1}(k_j)\}$.

4) N_i verifie si $h(h^{z-v-1}(k_j)) = h^{z-v}(k_j)$ et aussi vérifie si $h(Ax_j||h^{z-v-1}(k_j)) = s_j$ et $h(Kx_{ij}||h^{z-v-1}(k_j)) = z_j$. Si c'est valide, N_j est authentifié aussi par N_i.

5) N_i et N_j mettent à jour leur fonction de hachage respectivement à $h^{z-u-1}(k_i)$ et $h^{z-v-1}(k_j)$ et informent les autres nœuds du groupe via la station de base.

Phase d'ajout de nouveau nœud : la SB génère et déploie k_{r+1} et la chaine de hachage $h^z(k_{r+1})$ dans le nouveau nœud N_{r+1}. Puis informe les autres nœuds du réseau de $h^z(k_{r+1})$ et *z*. L'authentification et l'établissement de clés est la même que la phase précédente.

Cependant H. Sung, et al. [3], démontre que la méthode de H. Huang [1] est vulnérable à l'attaque par répétition et par mascarade et ne permet pas de renouveler la chaine de hachage. Par ailleurs, le paramètre *z* limite la durée de vie du réseau. Des vulnérabilités liées à l'attaque par mascarade du nouveau nœud dans sa phase d'addition de nouveau nœud sont toujours notées dans [4].

5.2. Authentification de requêtes

Pour séparer les concepts, nous distinguons deux types de requêtes : les requêtes des utilisateurs que nous appelons *requêtes externes* et celles du système c'est-à-dire, celles de la station de base ou des nœuds capteurs que nous appelons *requêtes internes*.

5.2.1. Authentification de requêtes externes

Dans ce cas, les propriétés de sureté et de vivacité sont définies comme suit :

- *sureté* : si capteur traite une requête, celle-ci est postée par un utilisateur légitime.

- *vivacité* : toute requête postée par un utilisateur légitime, sera traitée par les nœuds capteurs capables de fournir une réponse à la requête de cet utilisateur.

5.2.1.1. Réaliser l'Authentification Robuste de Requêtes d'Utilisateurs

L'idée de base de ce protocole proposé par Z.Benenson et al.[11], est de laisser les capteurs au voisinage de l'utilisateur servir d'interprètes entre la cryptographie asymétrique de l'utilisateur et celle symétrique des nœuds capteurs. L'utilisateur s'authentifie auprès des nœuds capteurs de son voisinage en utilisant la cryptographie asymétrique, et ensuite ces nœuds capteurs communiquent avec le reste du réseau en utilisant la cryptographie symétrique. La station de base sert d'autorité de certificat (*CA*) avec respectivement ses clés publique et privée *cle_privCA*, *cle_pubCA*. Le certificat d'un utilisateur (*U*) légitime est signé par le *CA* en utilisant la clé publique i.e. *certU = signCA(cle_pubU).* Chaque nœud capteur est pré-chargé avec la clé publique du *CA*, ce qui leur permet de vérifier indépendamment les certificats des utilisateurs. Puisque la cryptographie symétrique nécessite plus de calculs en déchiffrement et génération de signature qu'en chiffrement et vérification de signature, pour une authentification unilatérale d'un utilisateur, les nœuds capteurs du voisinage effectuent seulement la vérification de signature.

Ce protocole, par l'envoie de faux certificats, peut être vulnérable à l'attaque par brouillage à la couche MAC. Il ne traite pas les requêtes impliquant plusieurs nœuds capteurs, comme exemple : le calcul de la température dans une zone donnée du RCSF.

5.2.1.2. Authentification de Requêtes Basée sur la Clé Symétrique

Ce protocole de Banerjee et al.[12], improvise la technique de pré-distribution de clé de Blundo et al [5] fondée sur un polynôme bi-varié symétrique de degré *t*, et y ajoute un support additionnel d'authentification de requêtes. Il traite les requêtes d'un utilisateur impliquant plusieurs capteurs en utilisant la cryptographie symétrique. Seule la propriété de *sureté* est traitée, celle de vivacité est laissée à d'autres protocoles.

Dans ce protocole, un utilisateur diffuse son identité ID_u et sa requête *q*. Le réseau de capteurs identifie un ensemble de capteurs S_q capables de la traiter. Ces capteurs élient un leader. Ce dernier prend la responsabilité de générer un *nonce* (number once) qu'il va envoyer aux autres capteurs et qui sera notifié avec S_q à l'utilisateur. Pour chaque nœud capteur de S_q, l'utilisateur calcule un MAC, ensuite rassemble tous les MACs, puis les envoie à chaque capteur de S_q. Chaque capteur de S_q, après avoir reçu la collection de MACs, calcule un Mac sur le *nonce* et vérifie s'il y'a une correspondance dans la collection de MACs reçue. Si tel est le cas, la requête est acceptée.

Puisque le protocole est basé sur la méthode de Blundo et al [5], il est sécurisé à $(t-1)$ nombre de nœuds capturés ou *t-secure*, ce qui donne une garantie forte de sécurité contre la compromission des nœuds. Cependant, il n'est résistant à l'attaque déni de service par l'envoie de faux messages qui peuvent occuper un nœud capteur.

5.2.2. Authentification de requêtes internes

Les propriétés de *sureté* et de *vivacité* sont définit comme suit :

-*sureté :* si un nœud capteur traite une requête, celle-ci provient de la station de base ou d'un nœud capteur du réseau.

- *vivacité :* toute requête légitime sera reçue par tous les capteurs du réseau concernés.

Des solutions d'authentification de requêtes internes existent. Zinaida Benenson et al. proposent dans [14], une solution d'authentification des requêtes diffusées par la station de base(SB) dans le réseaux de capteurs. Cette solution, montre comment la SB authentifie ses requêtes auprès des nœuds capteurs tout en limitant la propagation de fausses requêtes. Elle utilise la cryptographie symétrique et est basée sur la méthode de Canetti et al. [6], mais plus performante et est liée à une coopération implicite entre les nœuds. Le protocole utilise la *stratégie de passe*, c'est-à-dire si un nœud capteur n'est pas en mesure de déterminer si une requête est légitime, il la passe à ses voisins. Un MAC de 1-bit est appliqué après une fonction de hachage. Cette idée est de [6].

D'autres approches comme SPINS (Security Protocole for Sensor Network) [13] à travers µTESLA réalisent l'authentification des requêtes diffusées en utilisant les fonctions de hachage, une clé symétrique et un temps de synchronisation. SPINS réalise l'authentification de la diffusion par l'asymétrie en utilisant un MAC.

5.3. Authentification des utilisateurs

Ces solutions sont moins adressées et utilisent des techniques basées sur un mot de passe fort ou faible. Les solutions d'authentification par mot de passe fort utilisent les fonctions de hachage et l'opérateur OU-exclusif, et sont utilisables dans les RCSFs.

La méthode proposée par Wong et al. [7], est basée sur le mécanisme de Lee et al. [8], et permet aux utilisateurs légitimes d'accéder au RCSF. Elle est composée de trois phases : une phase d'enregistrement, une phase login, et une phase d'authentification.

Phase d'enregistrement : l'utilisateur envoie son ID_u et son mot de passe *PW* à la passerelle du réseau *GW*. Le *GW* calcule des valeurs *A* et *B* puis enregistre ID_u et *PW*. Il distribue ID_u, *A*, et une estampille *Ts*, aux nœuds login (*LN*) qui sont en mesure de produire une interface login aux utilisateurs.

Phase login : l'utilisateur envoie sont ID_u et *PW* au *LN*. Si ID_u est valide, le *LN* récupère *A* et calcule non seulement *B*, mais aussi C_2 et C_1 puis envoie l'identité ID_u, C_2, C_1 l'estampille temporelle (Ts) à *GW* pour une authentification finale.

Phase d'authentification : le *GW* vérifie la validité de l'utilisateur et de l'estampille, récupère *A* et *B*, puis calcule C_1 et C_2.

Cependant des travaux ont montré que le protocole est vulnérable à l'attaque par répétition. Une amélioration est proposée dans Tseng et al. [9], celle-ci propose une quatrième phase de changement de mot de passe. Vaidya et al. [10] montrent que cette dernière solution reste vulnérable à l'attaque « man in the middle » (MITM) et font une proposition qui sera plus tard modifiée. Leur solution est résistante à l'attaque MITM et

Produit une authentification mutuelle aussi bien entre le *LN* et le *GW* qu'entre le *GW* et l'utilisateur.

Solutions	**Clés**	**Scalabilité**	**Complexité communication**	**Topologie**
Wong et al. [7],	Symétrique	Oui	$7T_H+4T_{XOR}+3C_{MH}=O(1)$	Hierarchique
Tseng *et* al. [9]	Symétrique	Oui	$5T_H+4T_{XOR}+3C_{MH}=O(1)$	Hierarchique
Vaidya et al. [10]	Symétrique	Oui	$11T_H+4T_{XOR}+3C_{MH}=O(1)$	hierarchique
Z.Benenson et al [11]	Asymétrique	Non	$2nT_H+3T_{EXP}=O(n)$	plate
Banerjee et al [12]	Symétrique	Non	$O(\sqrt{N})$	plate
H. Huang [1]	Asymétrique	Oui	$2T_{EM}+5T_H=O(1)$	Plate
H. Sung, et al[3]	Asymétrique	Oui	$2T_{EM}+8T_H=O(1)$	Plate

Tableau 1 : comparaison des différentes solutions

6. Comparaison des différentes solutions

Le *tableau 1* résume les différentes solutions avec quelques paramètres de comparaison. Quatre paramètres sont évalués. Le premier paramètre est *type de clé* qui peut être symétrique ou asymétrique. On constate que les protocoles d'ajout de nouveaux nœuds sont basés sur la cryptographie asymétrique qui n'est pas adéquate dans l'environnement des RCSF. Ils impliquent des capteurs homogènes, ce qui permettra de les adapter à la topologie plate. Par contre les protocoles d'authentification d'utilisateurs utilisent la cryptographie symétrique. Le deuxième est la *scalabilité* qui montre que le mécanisme passe bien à l'échelle. Dans le tableau, seuls les mécanismes d'authentification de requêtes externes ne passent pas à l'échelle. Par ailleurs il faut noter que les capteurs sont déployés dans des endroits peu sûrs, les protocoles de contrôle d'accès doivent utiliser une approche distribuée c'est-à-dire qu'une requête ou un nouveau capteur soit respectivement traitée ou ajouté par plusieurs capteurs afin d'éviter les problèmes de nœuds compromis ou capturés, ce qui est réalisé par ces protocoles car ils sont *t-secure,* et peuvent s'adapter à la topologie plate. La *complexité de communication* liée à la densité du réseau. Dans ce tableau, *n* dénote le nombre de nœuds au voisinage de l'utilisateur, *N* la taille du réseau, T_H, le temps pour l'exécution d'une fonction de hachage, T_{XOR} le temps pour effectuer une opération Ou-exclusif, C_{MH} le délai de communication multi-sauts entre le *LN* et le *GW*, T_{EXP} le temps pour calculer

une exponentiel modulaire, T_{EM} point de multiplication sur la courbe elliptique. La complexité de communication définit à la fois les communications entre entités, les opérations effectuées et les entités impliquées pour établir un mécanisme de contrôle d'accès. Dans certains mécanismes, elle dépend de la taille du réseau *N* ou du nombre de voisins *n*. Donc elle varie en fonction de la densité du réseau, c'est le cas des mécanismes d'authentification de requêtes externes. Par contre dans d'autres méthodes, elle est indépendante de la densité ou du nombre de voisins quand elle est à *O(1)*. Enfin le dernier paramètre est la topologie qui est soit plate ou hiérarchique. Dans les protocoles d'authentification d'utilisateurs, on a deux types de nœuds : des nœuds capteur *simples*, et des nœuds de *service* avec plus de puissance et destinés à effectuer plus d'opérations. Ces protocoles peuvent être adaptés à la topologie hiérarchique où les nœuds de services pourront servir de CH. Par contre dans d'autres protocoles, tous les nœuds capteurs sont identiques et ont les mêmes fonctions. Ce qui permettra leur adaptation à la topologie plate.

7. Conclusion

Plusieurs solutions sont basées sur la cryptographie asymétrique et résistent difficilement à l'attaque par répétition ou déni de service. On note que également beaucoup de solutions sont basées sur un mécanisme de pré-distribution de clés, d' une fonction de hachage et/ou de génération de nombre aléatoire. Par conséquent, le mécanisme de contrôle d'accès hérite toutes les vulnérabilités du mécanisme de pré-distribution sur lequel il repose. On peut envisager pour des travaux futurs, un mécanisme de contrôle d'accès intégrant à la fois ce mécanisme de pré-distribution.

Bibliographie

[1] H. F. Huang, "A novel access control protocol for secure sensor networks," Computer Standards & Interfaces, vol. 31, pp. 272-276, 2009.

[2] Y. Zhou, Y. Zhang, and Y. Fang, "Access control in wireless sensor networks", Ad Hoc Networks, Vol. 5, pp. 3-13, 2007.

[3] H. S. Kim and S. W. Lee, "Enhanced novel access control protocol over wireless sensor networks," IEEE Trans. Consum. Electron., vol. 55, no. 2, pp. 492- 498, 2009.

[4] P. Zeng, K-K..R Choo, D. Sun, "On the Security of an Enhanced Novel Access Control Protocol for Wireless Sensor Networks", IEEE Transactions on Consumer Electronics, Vol. 56, No. 2, May 2010, pages 566-569

[5] C. Blundo et al. "Perfectly-secure key distribution for dynamic conferences", in Advances in Cryptology CRYPTO 92, LNCS, 1993, pp. 471-486.

[6] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas. Multicast security: A taxonomy and some efficient constructions. In Proc. IEEE INFOCOM'99, volume 2, pages 708–716, New York, NY, Mar. 1999. IEEE.

[7] K.H.M. Wong, Y. Zheng, J. Cao, and S. Wang, "A dynamic user authentication scheme for wireless sensor networks." In Proc. of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06), Jun. 2006; 1: 318–327.

[8] C.Y. Lee, C.H. Lin, and C.C. Chang, "An Improved Low Communication Cost User Authentication Scheme for Mobile Communication", Proceedings of the IEEE 19 th International Conference on Advanced Information Networking and Applications (AINA 2005), Taiwan, March 2005.,

[9]Tseng, H. R., Jan, R. H., and Yang, W. 2007. An improved dynamic user authentication scheme for wireless sensor networks. In Proceedings of the IEEE Global Communications Conference (GLOBECOM'07), Nov. 2007;986-990.

[10] B Vaidya, J.S. Silva, J.J. Rodrigues, "Robust Dynamic User Authentication Scheme for Wireless Sensor Networks", *In Proc. Of the 5th ACM Symposium on QoS and Security for wireless and mobilenetworks (Q2SWinet 2009),*

[11]Z. Benenson, N. Gedicke, and O. Raivio "Realizing Robust User Authentication in Sensor Networks." In Proc. of Workshop on Real-World Wireless Sensor Networks (REALWSN 2005), Sweden, June 2005.

[12] S. Banerjee,D. Mukhopadhyay, "Symmetric Key Based Authenticated Querying In Wireless Sensor Networks", InterSense '06, Proceedings of the First International Conference on Integrated Internet Ad hoc and Sensor Networks, May 2006, Nice France

[13] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler. SPINS: security protocols for sensor networks. Wireless Networks, 8(5):521–534, 2002.

[14] Z. Benenson, L. Pimenidis, F. C. Freiling, and S. Lucks. Authenticated query flooding in sensor networks. In PERCOMW '06: Proceedings of the Fourth Annual IEEE International Conference on Pervasive Com-puting and Communications Workshops, page 644, Washington, DC,USA, 2006. IEEE Computer Society.